



varian data machines / a varian subsidiary

VARIAN 70 SERIES
MEMORY MAP MANUAL





**VARIAN 70 SERIES
MEMORY MAP MANUAL**

Specifications Subject to Change Without Notice



varian data machines / a varian subsidiary
2722 michelson drive, irvine, california 92664
© 1974 printed in USA



varian data machines

98 A 9906 100

February 1974



TABLE OF CONTENTS

SECTION I GENERAL DESCRIPTION

SECTION 2 INSTALLATION

2.1	INSPECTION	2-1
2.2	PHYSICAL DESCRIPTION	2-1
2.3	DISCRETIONARY WIRING	2-1
2.4	INTERCONNECTION	2-2
2.5	CONFIGURATIONS	2-2
2.5.1	Normal Configuration	2-2
2.5.2	WCS Supported Configuration A	2-2
2.5.3	WCS Supported Configuration B	2-3
2.5.4	System Memory Lockout Configuration	2-3
2.5.5	Dual Memory Map Configuration	2-3

SECTION 3 OPERATION

3.1	I/O INSTRUCTIONS	3-1
3.2	OPERATING MODES	3-2
3.2.1	Inactive Mode	3-2
3.2.2	Executive Mode	3-2
3.2.3	User Mode	3-3
3.3	MAPPING	3-3
3.4	OPERATING SEQUENCES	3-3
3.4.1	Memory-Map Loading and Read-Back	3-3
3.4.2	Programmed I/O Read-Back	3-3
3.4.3	Executive Mode to Inactive Mode	3-6
3.4.4	Inactive Mode to Executive Mode	3-6
3.4.5	Executive Mode to User Mode	3-6
3.4.6	User Mode to Executive Mode	3-6
3.5	ACCESS-CONTROL MODES	3-8
3.6	MEMORY PROTECTION	3-8
3.6.1	Halt Errors	3-8
3.6.2	I/O Errors	3-8
3.6.3	Writing Errors	3-8
3.6.4	Jump Errors	3-9
3.6.5	Unassigned Errors	3-9
3.6.6	Instruction-Fetch Errors	3-9
3.6.7	I/O Data-Transfer Errors	3-9



SECTION 4 THEORY OF OPERATION

4.1	GENERAL.....	4 1
4.2	SYSTEM DESCRIPTION.....	4 1
4.3	FUNCTIONAL CIRCUITS	4 1
4.3.1	Key Multiplexor.....	4-5
4.3.2	RAM Address Multiplexor.....	4-5
4.3.3	Drivers and Row-Selection Decoder.....	4-5
4.3.4	RAM Array.....	4-6
4.3.5	I/O-Bus Data Multiplexor.....	4-6
4.3.6	I/O-Bus Drivers and Receivers.....	4-7
4.3.7	Word-Transfer Counter.....	4-8
4.3.8	Key and 64K-Mode Registers	4-8
4.3.9	Executive-State Register	4-9
4.3.10	Map-Address Counter.....	4-9
4.3.11	Input-Data Selection Register.....	4-9
4.3.12	DMA I/O Register.....	4-9
4.3.13	DMA Memory-Address Counter.....	4-9
4.3.14	I/O-Error Key Register	4-10
4.3.15	Instruction Address Register	4-10
4.3.16	Unassigned Address Register.....	4-10
4.3.17	Swapping Control	4-10
4.3.18	Access Control	4-10
4.3.19	Memory Address Drivers and Receivers.....	4-11
4.3.20	Active/Inactive Register	4-11
4.3.21	User/Executive Mode Register.....	4-11
4.3.22	Bypass Drivers.....	4-11
4.3.23	Memory-Data Circuits.....	4-11
4.3.24	Address Detection and Function Control.....	4-12
4.3.25	Memory Control.....	4-13
4.3.26	DMA Control.....	4-13
4.3.27	Mode Switching Control.....	4-14
4.3.28	Interrupt Control	4-14
4.4	MEMORY MAP TIMING	4-15

SECTION 5 MAINTENANCE

5.1	TEST EQUIPMENT	5-1
5.2	CIRCUIT BOARD REPAIR	5-1
5.3	CIRCUIT-COMPONENT IDENTIFICATION.....	5-1



SECTION 6 MNEMONICS

SECTION 7 TEST PROGRAMS

7.1	GENERAL.....	7-1
7.2	TEST PROGRAM ORGANIZATION.....	7-1
7.2.1	Miscellaneous Register Test	7-1
7.2.2	Map Register Test.....	7-1
7.2.3	Executive-State Access and Map-Selection Test	7-1
7.2.4	Memory Test.....	7-1
7.2.5	Memory Protection Test.....	7-1
7.2.6	Utility Routines.....	7-1
7.3	PROGRAM-TAPE LOADING.....	7-2
7.4	SENSE SWITCHES.....	7-2
7.5	OPERATING PROCEDURES.....	7-2
7.6	INPUT DIRECTIVES FOR UTILITY ROUTINES.....	7-3
7.7	ERROR MESSAGES.....	7-3
7.7.1	Miscellaneous Register Test	7-4
7.7.2	Map Register Test.....	7-4
7.7.3	Executive-State Access and Map-Selection Test.....	7-4
7.7.4	Memory Test.....	7-4
7.7.5	Memory Protection Test.....	7-5
7.8	SUMMARY OF I/O MESSAGES.....	7-7
7.9	MAINTAIN II OPERATION WITH MEMORY MAP.....	7-7
7.9.1	Unmapped Environment.....	7-8
7.9.2	Mapped Environment.....	7-8
7.9.3	Example.....	7-8
7.10	TTY MESSAGE EXAMPLES.....	7-9



LIST OF ILLUSTRATIONS

Figure 1-1. Memory-Map Address Formation	1-2
Figure 2-1. Memory Map Board.....	2-1
Figure 2-2. Memory Map Interconnection.....	2-2
Figure 3-1. Data-Word Formats For Output-Data Transfers.....	3-4
Figure 3-2. Data-Word Formats For Input-Data Transfers	3-5
Figure 3-3. Data-Word Format For Memory-Map Loading and Read-Back Operations.....	3-6
Figure 3-4. User Mode to Execute Mode Flow Chart	3-7
Figure 4-1. Memory-Map System Block Diagram	4-1
Figure 4-2. Key Multiplexor Block Diagram	4-3 / 4-4
Figure 4-3. Memory Map Functional Block Diagram	4-5
Figure 4-4. Layout of RAM Array	4-6
Figure 4-5. RAM Array Block Diagram	4-7
Figure 4-6. I/O-Bus Data Multiplexor.....	4-7
Figure 4-7. I/O Bus Drivers	4-8
Figure 4-8. Executive-State Register Block Diagram	4-9
Figure 4-9. DMA I/O Register.....	4-9
Figure 4-10. Memory-Data Paths Between Processor and Expansion Memories	4-11
Figure 4-11. Data Flow for Map Loading Operation.....	4-13
Figure 4-12. Data Flow for Map Read-Back Operation	4-14
Figure 4-13. Programmed I/O Data Transfer.....	4-15
Figure 4-14. Memory-Map Loading via High-Speed DMA	4-16
Figure 4-15. Memory-Map Read-Back via High-Speed DMA.....	4-17
Figure 4-16. Memory Map Loading/Read-Back Termination	4-18
Figure 4-17. Memory Mapping.....	4-19
Figure 4-18. I/O and Halt Error Detection.....	4-20
Figure 4-19. Jump-Error Detection	4-20
Figure 4-20. Unassigned and Writing Error Detection	4-21
Figure 4-21. Instruction-Fetch Error Detection	4-21
Figure 4-22. I/O Data-Transfer Error Detection	4-22
Figure 4-23. Memory Protection Interrupt.....	4-22
Figure 4-24. Executive-Mode to Inactive-Mode Switching.....	4-23
Figure 4-25. Inactive-Mode to Executive-Mode Switching.....	4-23
Figure 4-26. Executive-Mode to User-Mode Switching.....	4-24
Figure 4-27. User-Mode to Executive-Mode Switching.....	4-24



LIST OF TABLES

Table 1-1. Memory Map Specifications.....	1-1
Table 1-2. Glossary	1-3
Table 3-1. I/O Instructions	3-1
Table 3-2. Executive-Mode States	3-3
Table 3-3. Access-Control Modes	3-8
Table 3-4. Interrupt Addresses	3-8
Table 4-1. Row-Selection Decoder Truth Table	4-6
Table 4-2. First Multiplexor Truth Table.....	4-8
Table 4-3. Second Multiplexor Truth Table	4-8
Table 4-4. Interrupt-Address Bit Configurations.....	4-8
Table 4-5. Access-Control Decoder Truth Table	4-10
Table 4-6. Memory-Data Multiplexor Truth Table.....	4-12
Table 4-7. EXC2 Decoder Truth Table.....	4-12
Table 4-8. Function Codes for SEN Instructions.....	4-12
Table 4-9. Output-Transfer Format Truth Table.....	4-13
Table 7-1. SENSE Switch Settings.....	7-2
Table 7-2. Test Number Assignments.....	7-3
Table 7-3. Memory Protection Error Descriptions.....	7-5



varian data machines



SECTION 1

GENERAL DESCRIPTION

The **Varian 70 Series Memory Map Manual** describes the memory map and its interface with Varian 70 series computers.

The manual is divided into seven sections:

- Introduction to the memory map, related publications, specifications, and glossary
- Installation and interconnection data
- Operation
- Theory of operation
- Maintenance
- Mnemonics list
- Test programs

There is also a volume 2 to this manual. Volume 2 is assembled when the hardware is shipped, and reflects the configuration of a specific system. It contains engineering documents, such as logic and installation drawings.

The following list contains the part numbers of other manuals pertinent to the Varian 70 series computers (the x at the end of each document part number is the revision number and can be any digit 0 through 9):

System Handbook	98 A 9906 01x
Processor Manual	98 A 9906 02x
Core Memory Manual	98 A 9906 03x
Semiconductor Memory Manual	98 A 9906 04x
Option Board Manual	98 A 9906 05x
Power Supply Manual	98 A 9906 06x
Microprogramming Guide	98 A 9906 07x
Writable Control Store Manual	98 A 9906 08x
Test Programs Manual	98 A 9952 06x
VORTEX II Reference Manual	98 A 9952 24x

The memory map performs address relocation and memory protection for up to 256K words ($K = 1,024$) of physical memory by translating the 16-bit memory address and a 4-bit key into an 18-bit physical address. Mapping operations can be performed independently in up to sixteen 32K logical (virtual) memory areas. A 64K-mode of operation is available to provide eight 64K logical-memory areas. Map numbers 0 through 15 are used to identify the logical memory area, with map 0 being reserved for the VORTEX II operating system. The logical memory addresses are mapped into physical memory pages consisting of 512 words each. Page assignments for each logical memory are under control of the VORTEX II page-allocation routine.

NOTE

Although the VORTEX II operating system is referred to in this manual, the memory map consists of general-purpose hardware that allows operation in other software environments.

Figure 1-1 is a block diagram showing the address translation that is performed by the memory map. Either the processor or priority memory access (PMA) option generates a 16-bit logical address plus a 4-bit key. The most-significant seven bits of the address are combined with the key bits to address a location in the memory map's random-access-memory (RAM) array. When the 64K-mode enabling signal is set by an I/O output-data transfer instruction, the memory map is placed in the 64K mode of operation. The RAM array produces a 13-bit output consisting of three fields:

- a. A 9-bit field that is concatenated with the least-significant 9-bit field of the logical address to form the 18-bit physical address.
- b. A 2-bit field used for access control.
- c. A 2-bit field used for swapping control.

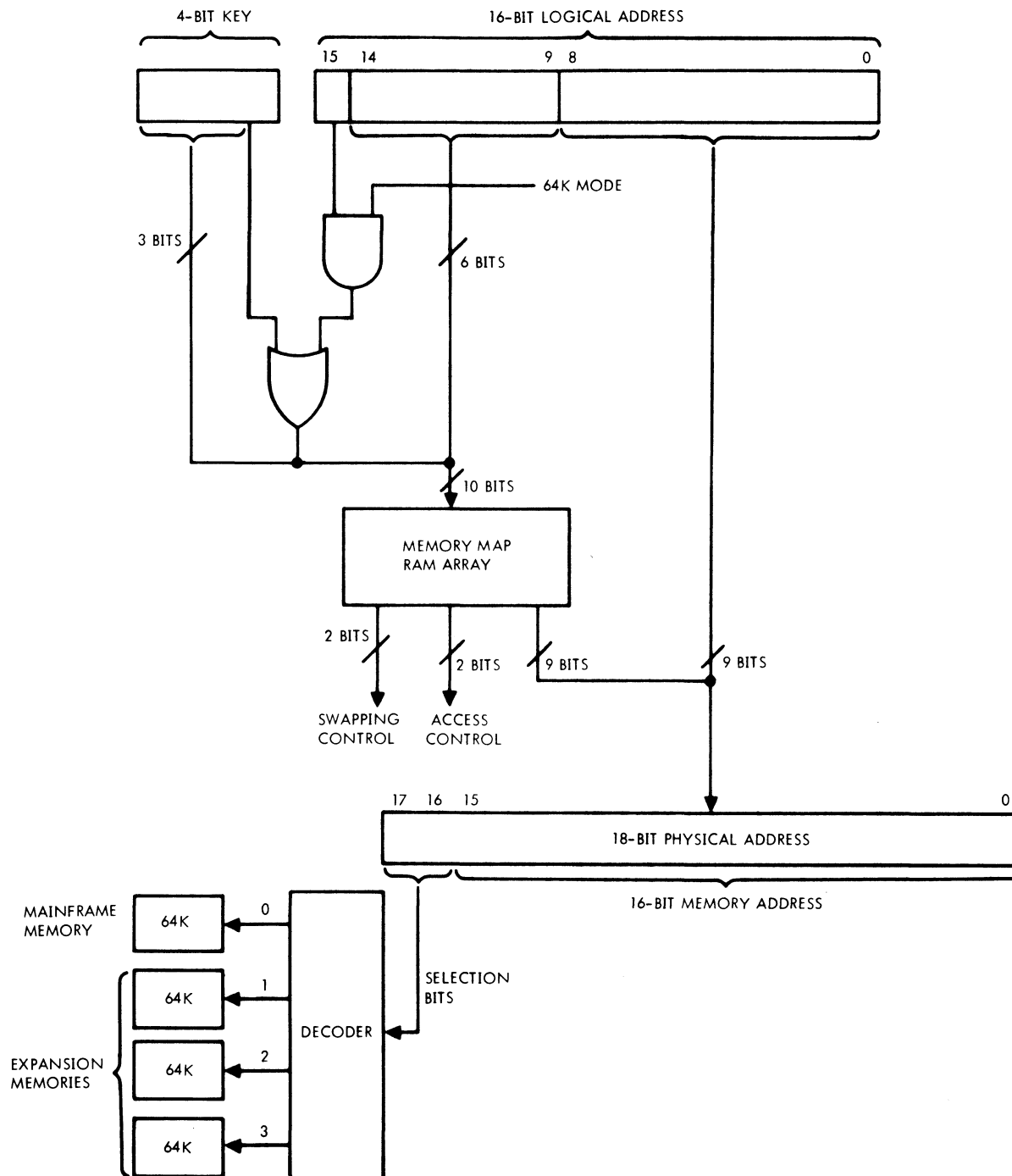
The most-significant two bits of the physical address are decoded to select one of the four 64K memory buses. The other 16-bits of the physical address are used to address a location in the selected memory module.

Specifications for the memory map are listed in table 1-1.

Table 1-1. Memory Map Specifications

Parameter	Specification
Physical memory size	Up to 256K words.
Logical memory size	Two modes are available: a. up to 32K words b. up to 64K words
Page size	512 words.
Number of logical memory areas	Up to 16 with 32K words. Up to 8 with 64K words. A combination of 32K and 64K word sizes is possible.

(continued)



VT11-2039

Figure 1-1. Memory-Map Address Formation

**Table 1-1. Memory Map Specifications** *(continued)*

Parameter	Specification
Memory access times	With the memory map active, memory access is delayed 104 nanoseconds for the first 64K of memory and 156 nanoseconds for memory above 64K. With the memory map inactive, memory access is delayed by 27 nanoseconds. These are worst-case delays for standard operating modes.
System configuration	Provides mapping of addresses for processor, DMA, and PMA on one memory port. Mapping on more than one port requires one memory map for each port (not supported by VORTEX II).
Loading	The memory-map RAM array is loaded and read via DMA operations. The loading word rate is 715 kHz; the reading word rate is 358 kHz.
Operating modes	User mode, executive mode, and inactive mode.
Priority assignments	The memory map's memory-protection feature is assigned the highest system priority. Priority assignment for DMA operation is made independently.
Logic levels (internal)	High = +2.4 to +5.0V dc Low = 0 to +0.4V dc
Logic levels (I/O bus)	High = +2.8 to +3.6V dc Low = 0 to +0.5V dc
Dimensions	Contained on a 15.6 by 19 inch printed-circuit board.
Installation	Plugs into a Varian 70 series mainframe chassis using one module slot.
Input power	+5V dc at 17 amperes.
Operational environment	0 to 50 degrees C, 0 to 90 percent relative humidity without condensation.

Table 1-2 is a glossary of terms used in this manual.

Table 1-2. Glossary

Term	Definition
Logical address	An address in a logical memory area.
Logical memory	A set of memory locations used by the programmer. Logical memory may or may not have contiguous physical memory locations.
Map numbers	Numbers 0 through 15 assigned to the maps used by the operating system and the various users. The numbers are determined by four key bits originating from either the BIC, PMA, map key register, or processor (using WCS microprogramming).
Mapping	The process of translating a logical memory address to a physical memory address.
Page	A 512-word block of physical memory.
Physical address	An address in physical memory.
Physical memory	Random-access memory defined by hardware.
Privileged instruction	Any instruction that causes a memory protection violation when used in the user mode (i.e., halt and I/O instructions). The halt instruction is only permitted in the inactive mode.
Swapping	The process of moving data between main and auxiliary memory in order to multiplex the use of main memory.



varian data machines



SECTION 2 INSTALLATION

2.1 INSPECTION

The memory map has been packed and inspected to ensure its arrival in good working order. To prevent damage, take care during unpacking and handling. Check the shipping list to ensure that all equipment has been received. Immediately after unpacking, inspect the equipment for shipping damage. Ascertain that wires and cables are neither loose nor broken, and that hardware is secure. If damage exists:

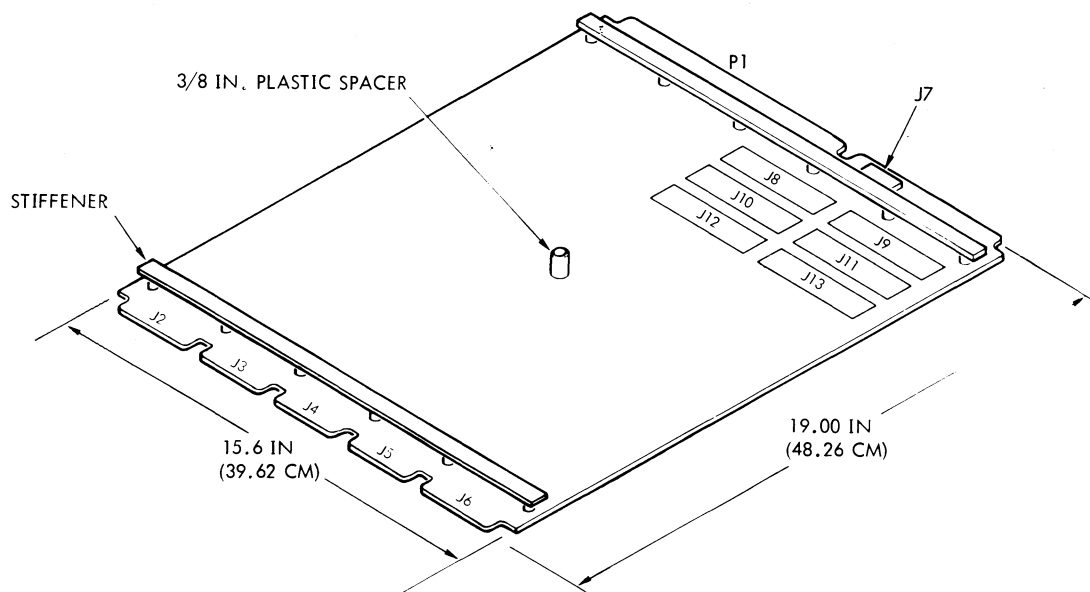
- a. Notify the transportation company.
- b. Notify Varian Data Machines.
- c. Save all packing material.

2.2 PHYSICAL DESCRIPTION

The memory map circuits are on a 15.6 by 19 inch printed-circuit (PC) board (p/n 44P0685). Figure 2-1 shows dimensions and connectors of the memory map board.

2.3 DISCRETIONARY WIRING

Connections of the various jumper terminals on the memory map PC board are listed in the memory map option drawing 01A1541 (volume 2). These connections are normally installed at the factory, but are referenced here in case the user wishes to have his memory map system expanded or changed in the field. The jumper-terminal designations referred to in the option drawing appear on the memory map board adjacent to the particular terminal.



VT11-2040

Figure 2-1. Memory Map Board



2.4 INTERCONNECTION

The memory map board plugs into a single module slot of the Varian 70 series mainframe chassis. Functions of the memory map board connectors are listed as follows:

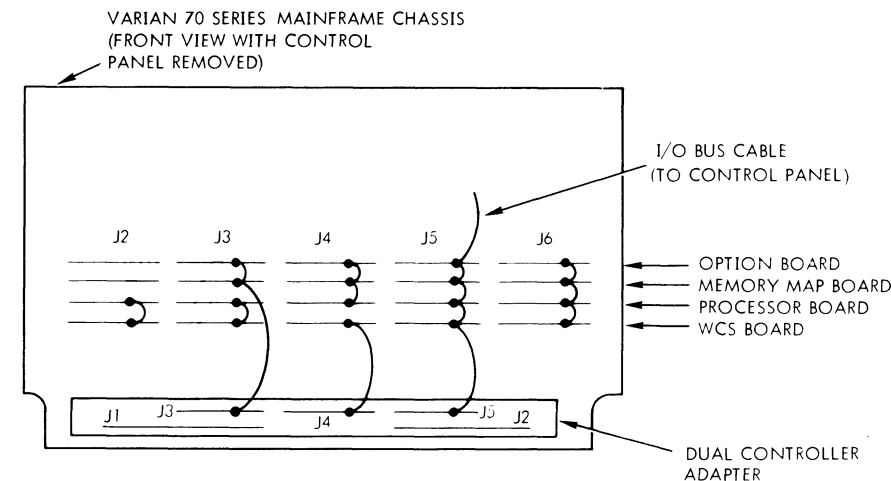
- P1, mainframe memory
- J2, not used
- J3, option-board auxiliary I/O bus
- J4, processor
- J5, I/O bus
- J6, option board and processor

J7, power

J8, through J13, expansion memories

Power supplied to the memory map board via connector J7 is normally provided by a +5-volt power supply (p/n 01P1280). It may also be supplied by excess +5-volt power from a memory expansion supply.

Figure 2-2 shows the memory map board interconnection in a Varian 70 series mainframe. Pin assignments for the connectors on the memory map board are provided in the logic diagram (p/n 91C0448) in volume 2 of this manual.



VTII-2041

Figure 2-2. Memory Map Interconnection

2.5 CONFIGURATIONS

The following subsections describe the memory-map configurations that are available.

2.5.1 Normal Configuration

The normal memory map configuration is supported by VORTEX II and has the following characteristics:

- a. The key bits are provided by the key register (section 4.3) on the memory map board.
- b. The active memory map is placed in the executive mode (section 3.2) on any interrupt.
- c. The memory map enters the user mode (section 3.2) by the EXC2 0246 instruction followed by a jump instruction (section 3.1).

- d. Privileged instructions are assigned to map 0.

2.5.2 WCS Supported Configuration A

In this configuration, the key register on the processor board is used (instead of the key register on the memory map board) to permit rapid key changes through microprogramming rather than key changes over the I/O bus. Characteristics of this configuration are:

- a. The key bits are provided by the key register on the processor board.
- b. The active memory map is placed in the executive mode on any interrupt.
- c. The memory map enters the user mode by the EXC2 0246 instruction followed by a jump instruction.



- d. Privileged instructions can be assigned to map 0 as in the normal configuration, or they can be assigned to all maps in which case there is no privileged instructions.

2.5.3 WCS Supported Configuration B

This configuration is the same as the A version except that interrupts are handled with microprogramming instead of the executive mode. In this configuration, the executive mode is jumper disabled. Characteristics of this configuration are:

- a. The key bits are provided by the key register on the processor board.
- b. With the executive mode disabled, there is no hardware distinction between the executive and user modes.
- c. The memory map enters the active mode by the EXC2 0146 instruction followed by a jump instruction.
- d. Privileged instructions can be assigned to map 0 as in the normal configuration, or they can be assigned to all maps in which case there is no privileged instructions.

2.5.4 System Memory Lockout Configuration

Through jumper connections on the memory map board, this configuration connects a memory lockout signal (MHGY- or MHMY-) to one of the expansion memories (MHGYn-(1-3) or MHMYn-(1-3)).

2.5.5 Dual Memory Map Configuration

In this configuration, two memory map boards are used, one for each memory port. Systems using this configuration contain a PMA and processor on different memory ports. Characteristics of this configuration are:

- a. The two memory maps are assigned the same device address.
- b. The two memory maps are loaded simultaneously from the I/O bus via high-speed DMA.
- c. The following functions are jumper disabled on the memory map connected to the PMA:
 - 1. Memory protection (except writing errors)
 - 2. Memory map read-back
 - 3. Executive mode
- d. To avoid possible conflicts in using the memory map, a software interlock is required between memory-map loading/read-back and PMA operations.



varian data machines



SECTION 3

OPERATION

The memory map contains no operating controls or indicators. Operation of the memory map is normally controlled by the VORTEX II operating system. However, by writing his own control program, the user can operate the memory map without using VORTEX II. For maintaining and testing the memory map, a MAINTAIN II memory map test program is available (section 7).

3.1 I/O INSTRUCTIONS

Memory-map I/O instructions can only be executed from map 0 (operating system) or an inactive memory map. Table 3-1 lists the I/O instructions with the mnemonics and octal codes for device address 46. An alternate device address is 56.

Table 3-1. I/O Instructions

Mnemonic	Octal Code	Function
External Control		
EXC2 046	104046	Executive Mode to Inactive Mode. Places the memory map in the inactive mode upon fetching the contents at the effective address of the jump instruction that follows.
EXC2 0146	104146	Inactive Mode to Executive Mode. Places the memory map in the executive mode upon fetching the contents at the effective address of the jump instruction that follows.
EXC2 0246	104246	Executive Mode to User Mode. Places the memory map in the user mode upon fetching the contents at the effective address of the jump instruction that follows.
EXC2 0346	104346	Start DMA Transfer. Starts a memory map DMA transfer.
EXC2 0446	104446	Reset DMA Transfer. Resets the memory map's DMA-transfer logic.
EXC2 0546	104546	Clear Executive-Mode Mask. Removes the executive-mode mask.
EXC2 0646	104646	Enable Memory Protection. Enables the memory protection function of the memory map.
EXC2 0746	104746	Disable Memory Protection. Disables the memory protection function of the memory map.
Sense		
SEN 046	102046	Sense DMA Activity. Senses if the memory map is performing a DMA operation.

(continued)



Table 3-1. I/O Instructions (continued)

Mnemonic	Octal Code	Function
SEN 0146	102146	Sense Abnormal DMA Termination. Senses for the error termination of a DMA loading or read-back operation.
Transfer*		
IME 046	102046	Transfers data from memory map to main memory.
INA 046	102146	Transfers data from the memory map to the A register.
INB 046	102246	Transfers data from the memory map to the B register.
CIA 046	102546	Transfers data from the memory map to the cleared A register.
CIB 046	102646	Transfers data from the memory map to the cleared B register.
OME 046	103046	Transfers data from main memory to the memory map.
OAR 046	103146	Transfers data from the A register to the memory map.
OBR 046	103246	Transfers data from the B register to the memory map.

*These transfer instructions are for control registers in the memory map. Loading and reading of the memory-map RAM array occur with the high-speed DMA operations.

3.2 OPERATING MODES

The memory map has three modes of operation: inactive, executive, and user.

3.2.1 Inactive Mode

When the memory map is in the inactive mode, the first 32K of physical memory (64K if the writable control store is used) is available unmapped and all instructions are permitted. This mode is entered either by a system reset condition or a branch sequence from the executive mode consisting of the EXC2 046 instruction followed by any jump instruction.

3.2.2 Executive Mode

This mode is entered by an interrupt or by a branch from the inactive mode to an active-map condition. The branch

sequence is actually an EXC instruction followed by any jump instruction. In this mode, all instructions except HLT are permitted. From the executive mode, the memory map can be switched to the inactive mode.

The executive mode has four states that define operations occurring between map 0 and the user maps. Setting up of the executive-mode states is accomplished with I/O instructions under control of the VORTEX II operating system. As shown in table 3-2, instruction-fetch operations are always from map 0, while operand-fetch and operand-store operations can be from any map depending on the executive-mode state. The following are exceptions:

- To ensure that all instruction fetches are from map 0, indirect addressing must not exceed the first level in states 2 or 3 of the executive mode. This is because after the first level of indirect addressing, instruction fetches in some cases (i.e., SRE and IJMP instructions) are treated as operand fetches by the memory map.

(continued)



- b. In all executive-mode states, the execution of a jump-and-mark instruction causes the program-counter contents to be stored in map 0.
- c. In all executive-mode states, the execution of a LDAI, LDBI, or LDXI instruction always causes the effective register to be loaded with the operand fetched from map 0.
- d. Any interrupt causes the memory map to enter the masked executive mode. This masked condition causes the executive mode to operate as if it were in state 0 even though the executive-state register may contain another value. In this condition, memory-map status is read into map 0 during the interrupt service routine. By executing the EXC2 0546 instruction, the mask is removed and the executive mode returns to the state determined by the value in the executive-state register.

Table 3-2. Executive-Mode States

State	Instruction Fetch	Operand Fetch	Operand Store
0	Map 0	Map 0	Map 0
1	Map 0	Map 0	Map n
2	Map 0	Map n	Map 0
3	Map 0	Map n	Map n

Notes:

1. Map 0 refers to the operating system.
2. Map n refers to the user map specified by the key bits.
3. States 1 through 3 cause an additional 142 nanoseconds delay in memory accessing.

3.2.3 User Mode

In this mode, all operands and instructions are mapped according to the key bits and contents of the RAM array. If an interrupt occurs, the memory map is switched to the executive mode. The memory map is switched from the executive mode to the user mode by the real-time executive in the VORTEX II operating system.

3.3 MAPPING

The mapping function is performed for either a DMA, PMA, or processor operation. In DMA operations, the mapped address utilizes the 4-bit key from the BIC plus the 16-bit logical address from the processor. In PMA operations, the mapped address utilizes the 4-bit key from the block-transfer controller (BTC) plus the 16-bit logical address from

the PMA option. In processor operations, the mapped address is normally derived from the memory map's key register and the 16-bit logical address from the processor.

3.4 OPERATING SEQUENCES

This section describes the sequences of the various memory-map operations. Figures 3-1 and 3-2 show the data-word formats used with output- and input-data transfers, respectively. Figure 3-3 shows the data-word format for the memory-map loading and read-back operations.

3.4.1 Memory-Map Loading and Read-Back

The following sequences occur in memory-map loading and read-back operations via DMA (using map 0):

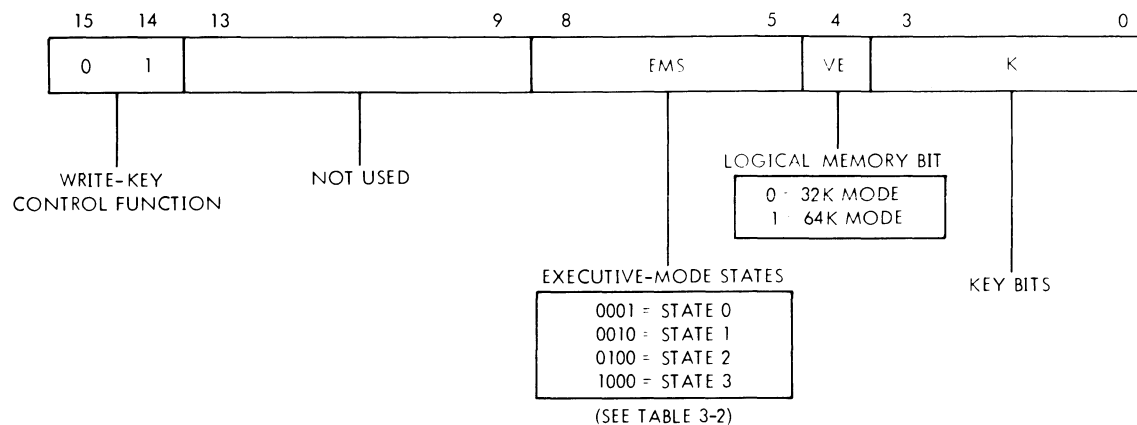
- a. Using output-data transfer instructions, (OME, OAR, or OBR), the processor transfers three data words to the memory map. The first word contains the direction of DMA transfer and the initial 10-bit map address (figure 3-1b). The second word contains the initial 16-bit memory address of the DMA transfer (figure 3-1c). The third word contains the number of words in the DMA transfer (figure 3-1d).
- b. The processor issues the EXC2 446 instruction to reset the DMA-control logic in the memory map.
- c. The processor issues the EXC2 0346 instruction to start the DMA transfer operation.
- d. The standard method for verifying the completion of DMA transfers consists of using the two SEN instructions. The processor issues SEN 046 to sense if the memory map is still performing the DMA transfer. If it is not performing the DMA transfer, SEN 0146 is issued to sense if the DMA-transfer termination is due to an error.
- e. An optional method for verifying the completion of DMA transfers consists of using the DMA-completion interrupt. A counter in the memory map counts the number of DMA transfers and indicates when all transfers are complete. An interrupt is sent to the processor when either all transfers are completed or an error occurs during one of the transfers. When the interrupt is acknowledged, the processor is directed to memory address 016.

3.4.2 Programmed I/O Read-Back

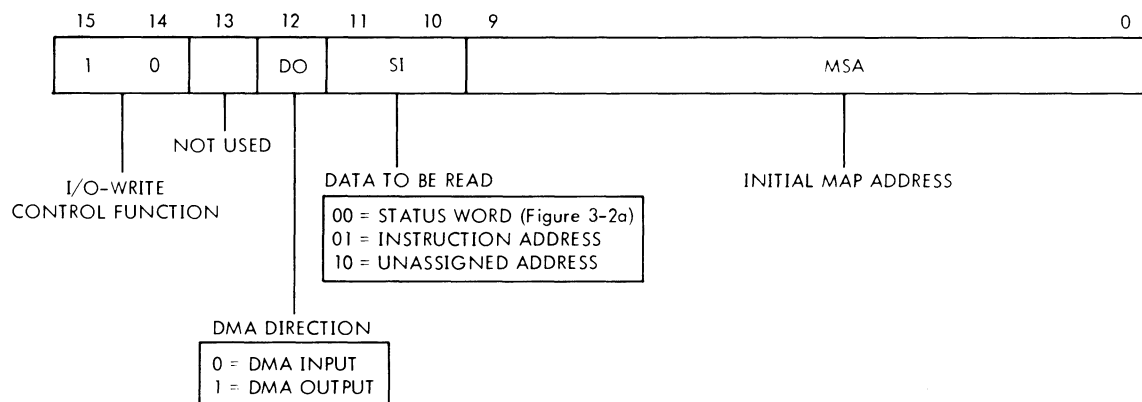
This operation provides a read-back function of the memory map's internal-status signals (figure 3-2a), instruction address register (figure 3-2b), and the unassigned



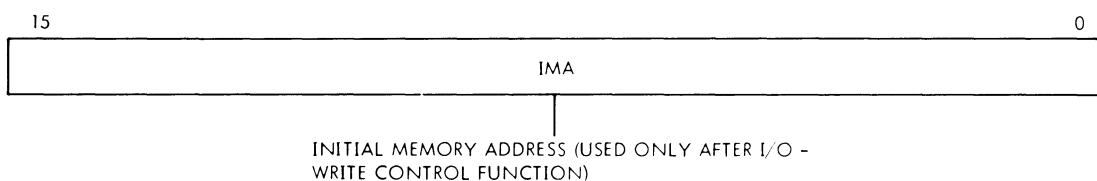
OPERATION



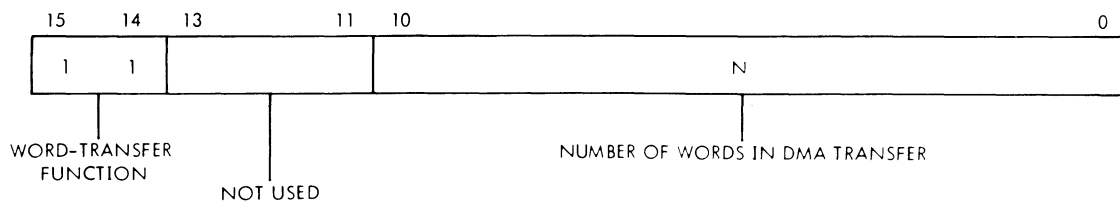
a. Write-Key Control Function



b. I/O-Write Control Function



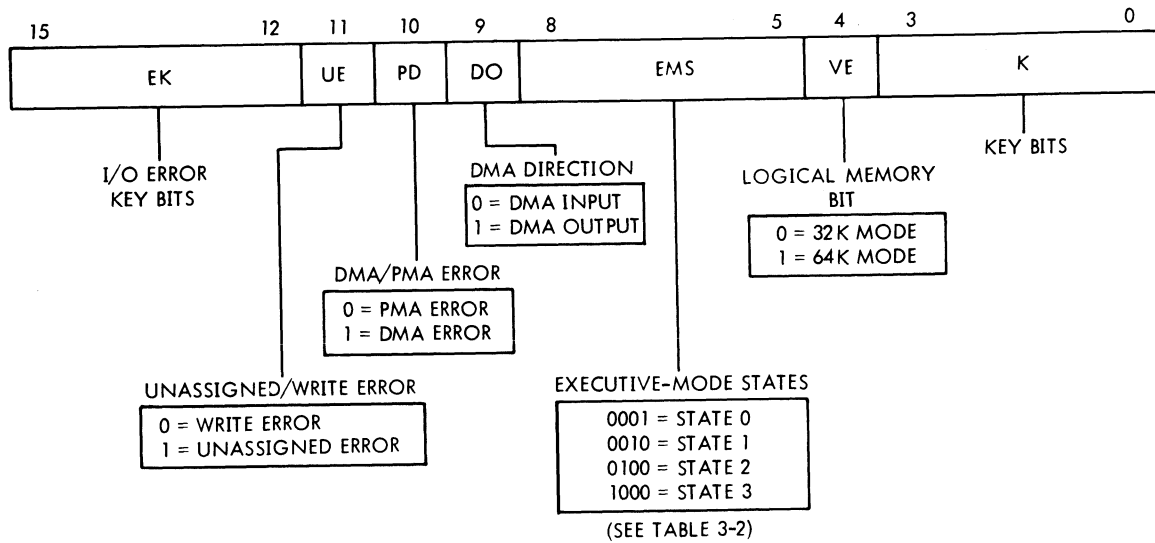
c. Initial Memory Address of a DMA Transfer



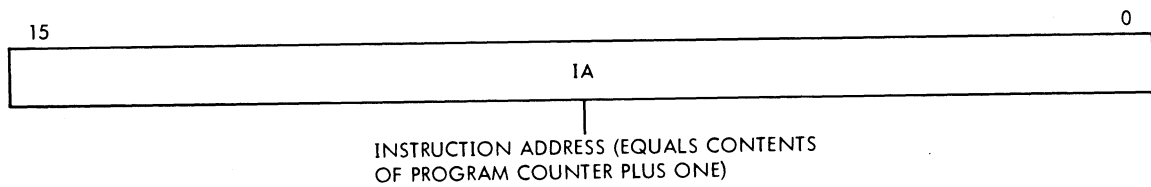
d. Word-Transfer Function

VT11-2042

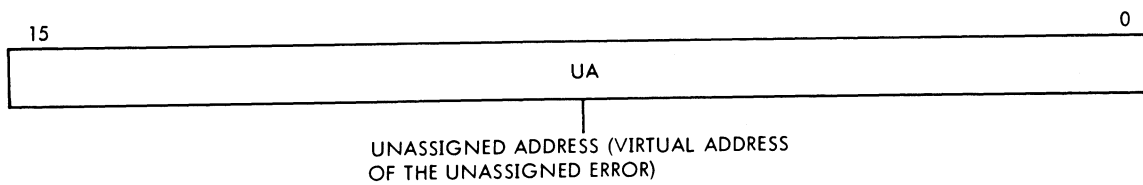
Figure 3-1. Data-Word Formats For Output-Data Transfers



a. Status Word



b. Instruction Address



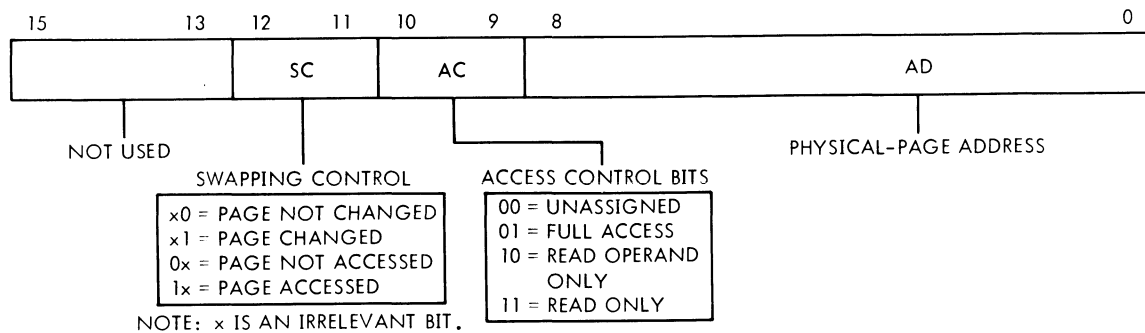
c. Unassigned Address

VTII-2045

Figure 3-2. Data-Word Formats For Input-Data Transfers



OPERATION



VTII-2047

Figure 3-3. Data-Word Format For Memory-Map Loading and Read-Back Operations

address register (figure 3-2c). The sequences of operations are listed below:

- Using an output-data transfer instruction, the processor transfers a data word (figure 3-1b) to the memory map. The type of data to be read back is specified by bits 10 and 11 of this data word.
- Using an input-data transfer instruction (IME, INA, INB, CIA, or CIB), the processor reads back the data specified by the output-data word.

- The processor executes a jump instruction. If the jump condition is not met, the memory map remains in the inactive mode.
- If the jump condition is met, the memory map switches to the executive mode when the contents of the effective jump address are fetched. The effective address is mapped using map 0.

3.4.3 Executive Mode to Inactive Mode

The following sequences occur when the memory map is switched from the executive mode to the inactive mode:

- The processor issues the EXC2 046 instruction to enable switching to the inactive mode.
- The processor executes a jump instruction. If the jump condition is not met, the memory map remains in the executive mode.
- If the jump condition is met, the memory map switches to the inactive mode when the contents of the effective jump address are fetched. The effective jump address is not mapped.

3.4.4 Inactive Mode to Executive Mode

The following sequences occur when the memory map is switched from the inactive mode to the executive mode:

- The processor issues the EXC2 0146 instruction to enable switching to the executive mode.

3.4.5 Executive Mode to User Mode

The following sequences occur when the memory map is switched from the executive mode to the user mode:

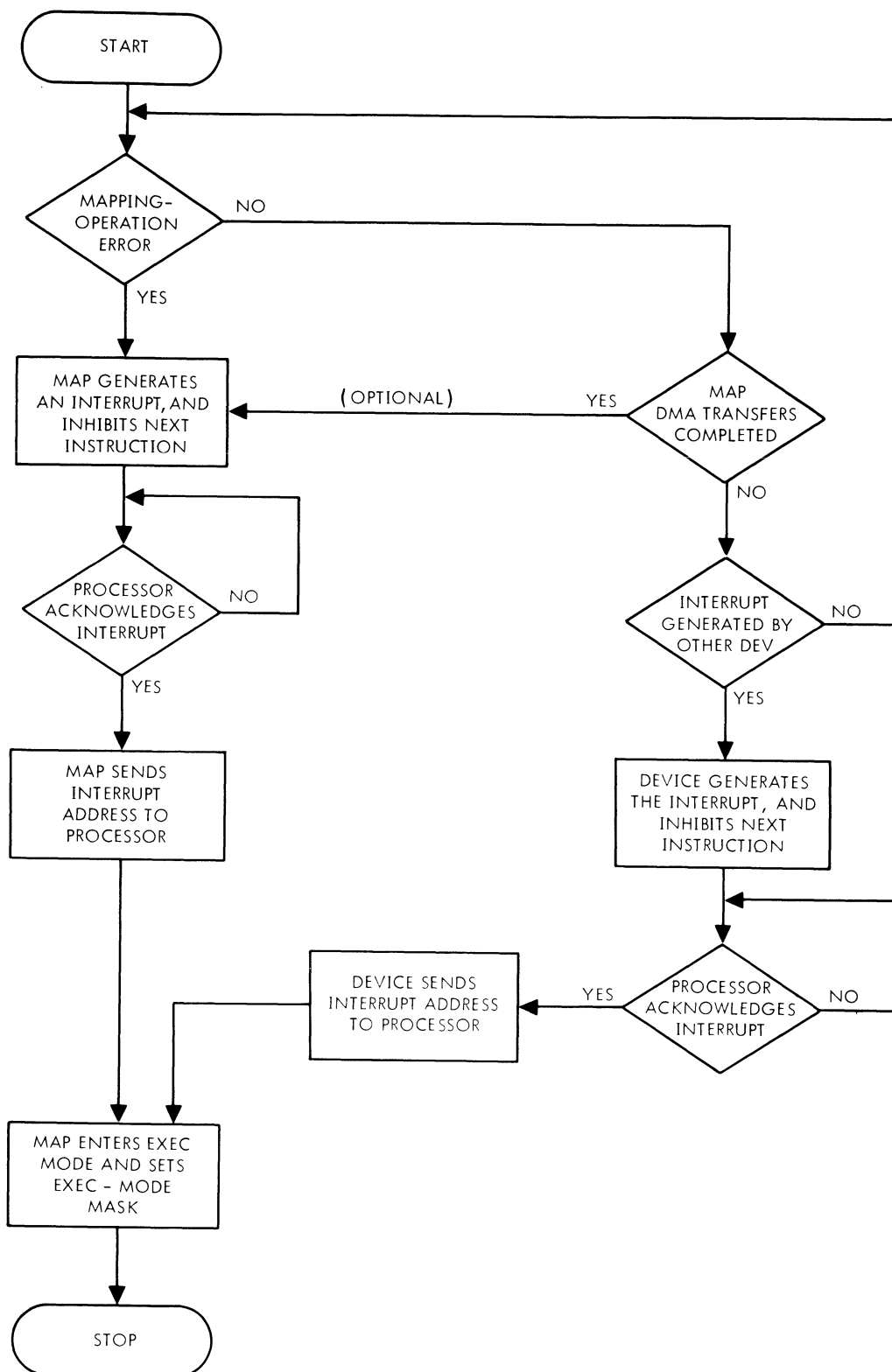
- The processor issues the EXC2 0246 instruction to enable switching to the user mode.
- The processor executes a jump instruction. If the jump condition is not met, the memory map remains in the executive mode.
- If the jump condition is met, the memory map switches to the user mode when the contents of the effective jump address are fetched. The effective address is mapped using the user map.

3.4.6 User Mode to Executive Mode

Switching the memory map from the user mode to the executive mode, can be initiated by interrupts resulting from the following:

- an error during a mapping operation
- an I/O system interrupt

The sequences of this operation are shown in the flow chart of figure 3-4.



VT11-2048

Figure 3-4. User Mode to Execute Mode Flow Chart



3.5 ACCESS-CONTROL MODES

Four access-control modes for mapping operations are provided by access-control bits 9 and 10 of the RAM-array data word. The four modes with corresponding binary values of bits 9 and 10 are listed in table 3-3.

Table 3-3. Access-Control Modes

Bits 10 9	Mode	Function
0 0	Unassigned	The logical address is unassigned (non-resident address).
0 1	Full access	All types of access are permitted in this page.
1 0	Read operand only	Only operand fetches are permitted. Instruction fetches from this page will not be executed. This restriction includes execution instructions (XEC, XOF, etc.).
1 1	Read only	Only instruction or operand fetches are permitted in this page (no operand-store operations permitted). The instruction fetches include single- and double-word instructions.

3.6 MEMORY PROTECTION

The memory protection function monitors the address of the instruction being processed on the basis of the access-control mode. When the memory map is active and its memory protection function is enabled, the instruction address register (in the memory map) is updated with each decoded instruction. When an error condition is detected, the memory protection function is disabled, and updating of the instruction-address register is inhibited until the EXC2 0646 instruction is executed. The detection of an error interrupts the program in process and directs it to one of seven preassigned memory addresses. These interrupt addresses are listed in table 3-4. Halt, I/O, and jump errors are detected earlier in the mapping operation to prevent them from being detected at the same time as the other errors.

Table 3-4. Interrupt Addresses

Octal Address	Error
20	Halt. The execution of a halt instruction is attempted.
22	I/O. The execution of an I/O instruction is attempted from a map number other than 0.

(continued)

Octal Address	Error
24	Write. An attempt is made to write into read-only or read-operand-only locations.
26	Jump. An attempt is made to jump into a read-operand-only location.
30	Unassigned. A read, write, or jump operation is attempted using an unassigned logical address.
32	Instruction fetch. An attempt is made to fetch an instruction from a read-operand-only location.
34	Data transfer. A write or unassigned error is detected during a DMA or PMA data transfer.

3.6.1 Halt Errors

When a halt error is detected, the execution of the halt instruction is allowed to be completed. However, due to the detection of the halt error, the memory map holds the memory-protection interrupt flag true (OINT-) so that the processor reenters the run mode immediately after halting. The processor then goes to an interrupt-wait state until the program being processed is directed to the interrupt address 20. Upon completion of the interrupt subroutine the interrupt signal OINT- is reset. A halt error is not detected when the halt is initiated manually using the STEP/RUN switch on the computer control panel.

3.6.2 I/O Errors

When an I/O error is detected, the execution of the I/O instruction is allowed to be completed. However, all I/O control functions and data transfers between the processor and peripheral controllers are inhibited. By holding the I/O-instruction error flag true, the contents of memory and the A, B, and X registers can not be modified by the I/O instruction. When the execution of the I/O instruction is completed, the program being processed is directed to the interrupt address 022.

3.6.3 Writing Errors

When a writing error is detected, the execution of the instruction is allowed to be completed. To prevent memory modification, the writing cycle for memory is changed to a reading cycle. When the execution of the instruction is completed, the program being processed is directed to the interrupt address 024.

Even though the writing error is not detected because the memory protection function is disabled, the active memory map always changes the writing sequence to a reading



sequence if an attempt is made to write into a non full-access location. This protection applies to all DMA, PMA (if connected to memory map), and processor memory cycles.

3.6.4 Jump Errors

A jump error can occur during the following types of instructions:

- a. All jump instructions including IJMP, JSR, and BT
- b. All jump-and-mark instructions including SRE

A jump error occurs when an attempt is made to jump or skip to a read-operand-only location and if this location is the effective address of the jump or skip instruction.

When a jump error is detected, the execution of the instruction is allowed to be completed. For jump-and-mark instructions, the memory writing cycle is changed to a reading cycle to prevent memory modification. When the execution of the instruction is completed, the program being processed is directed to the interrupt address 026.

3.6.5 Unassigned Errors

When an unassigned error is detected, the execution of the instruction is allowed to be completed. If the memory writing cycle contains the unassigned error, it is changed to a reading cycle. The contents of the A, B, and X registers are not changed. When the execution of the instruction is completed, the program being processed is directed to the interrupt address 030. The unassigned logical address is contained in the memory map's unassigned address register, which can be read by the processor using an input-data transfer I/O instruction (figure 3-2c).

3.6.6 Instruction-Fetch Errors

When an instruction-fetch error is detected, the execution of the current instruction is allowed to be completed. The program being processed is directed to the interrupt

address 032, and the next erroneous instruction is not executed.

3.6.7 I/O Data-Transfer Errors

The detection of an I/O data-transfer error during a DMA or PMA operation causes the memory map to generate an interrupt that directs the processor to the interrupt address 034.

When an I/O data-transfer error occurs during a DMA operation, the memory map holds the DMA termination signal (BIMES-I) true. This causes the BIC (or a user-designed controller) to terminate the data transfer on the trailing edge of the data-ready signal (DRYX-I or DRYF-I).

When a data-transfer error occurs during a PMA operation, the memory map holds the PMA termination signal (BTMES-I) true. This causes the block-transfer controller (or a user-designed controller) to terminate the data transfer immediately after receiving the termination signal.

When an I/O data-transfer error occurs, the memory map stores error-status data that include error-key number, writing or unassigned error, and DMA or PMA error. This error-status data (figure 3-2a) can be read by the processor by using an input-data transfer instruction.

When an I/O data-transfer error is detected during a memory-map DMA loading or read-back operation, the memory map generates a DMA error flag (PDMTRM+). In response to the SEN 046 instruction, the memory map provides a DMA not-busy status.

An optional mode of operation, for the detection of an I/O data-transfer error in a DMA loading or read-back operation, consists of using the DMA completion interrupt. When the error is detected, the memory map generates the interrupt directing the processor to the interrupt address 016. The same interrupt is also generated to signal the processor when the loading or read-back operation is successfully completed. Upon receipt of this interrupt, the processor can issue the SEN 0146 instruction to sense if the DMA transfer termination is due to an error.



varian data machines



SECTION 4

THEORY OF OPERATION

4.1 GENERAL

This section contains system and functional descriptions followed by timing waveforms for the various memory-map operations. For ease of reading, some mnemonics are written with the variable n in place of the actual numbers. For example, memory data mnemonics MYDA00- through MYDA17- are written MYDAn- (0-17). Mnemonic descriptions are provided in section 6.

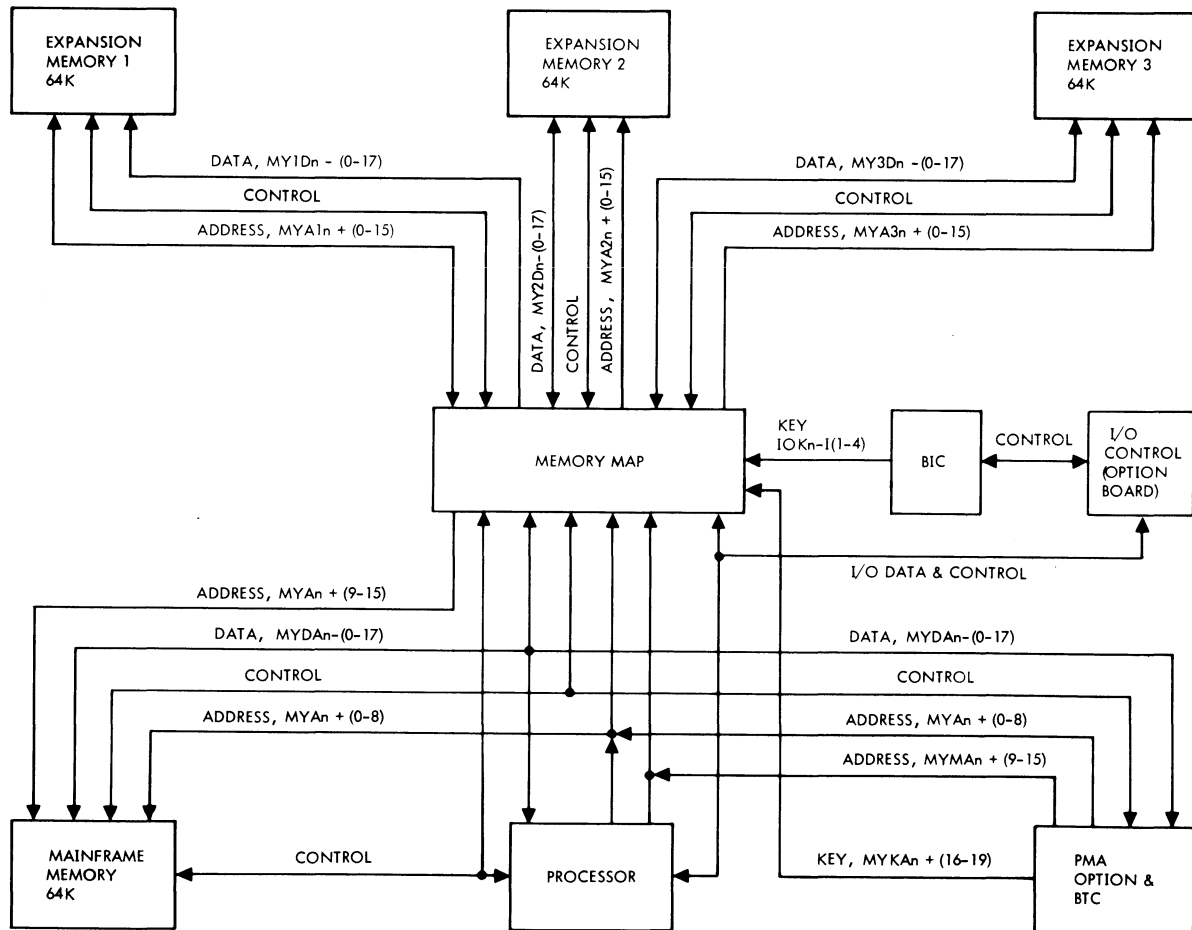
4.2 SYSTEM DESCRIPTION

As illustrated in the system block diagram of figure 4-1, the memory map can be used with either the processor or PMA

circuits. Address bits 0 through 8 are applied directly to the mainframe memory, while bits 9 through 15 are routed through the memory map. In addition to the mainframe-memory bus, the memory map provides buses for up to three expansion memories. The BIC provides I/O key bits for DMA memory mapping.

4.3 FUNCTIONAL CIRCUITS

The functional circuits of the memory map are shown in figure 4-2. Page numbers of the memory-map logic diagram (p/n 91C0448 in volume 2) are provided in parentheses for each circuit block.



VT11-2049

Figure 4-1. Memory-Map System Block Diagram



varian data machines



4-3 / 4-4



4.3.1 Key Multiplexor

The key multiplexor supplies the 4-bit key $PKMMX_n + (0-3)$ to the RAM address multiplexor. The 4-bit key is selected from one of the following sources:

- Key register, $PPRKY_n + (0-3)$
- BIC key, $IOK_n - I(0-3)$
- PMA key, $MYK_n + (16-19)$ or $MYKB_n + (16-19)$

As illustrated in figure 4-3, the key multiplexor contains two multiplexors, a latch, and a buffer. Depending on the jumper configuration, the processor key input of the first multiplexor comes from either the key register $PPRKY_n + (0-3)$ (standard for VORTEX II) or the latch $PMAKY_n + (0-3)$. PMA key bits are transferred through the latch when the PMA loading signal $PLPMAK +$ is low (this loading signal is low until half-clock time of a memory request). The other input of the first multiplexor is connected directly to the PMA key bits. A high $PKYMXE -$ disables the first multiplexor causing its output bits to be all low (map 0). A low $PKYMXE -$ enables one of the two inputs to be selected. Selection is controlled by $PMAKO +$. When the processor has accepted a PMA request, a high $PMAKO +$ transfers $PMYK_n + (16-19)$ through the first multiplexor. A low $PMAKO +$ selects the other input $PRMAK_n + (0-3)$.

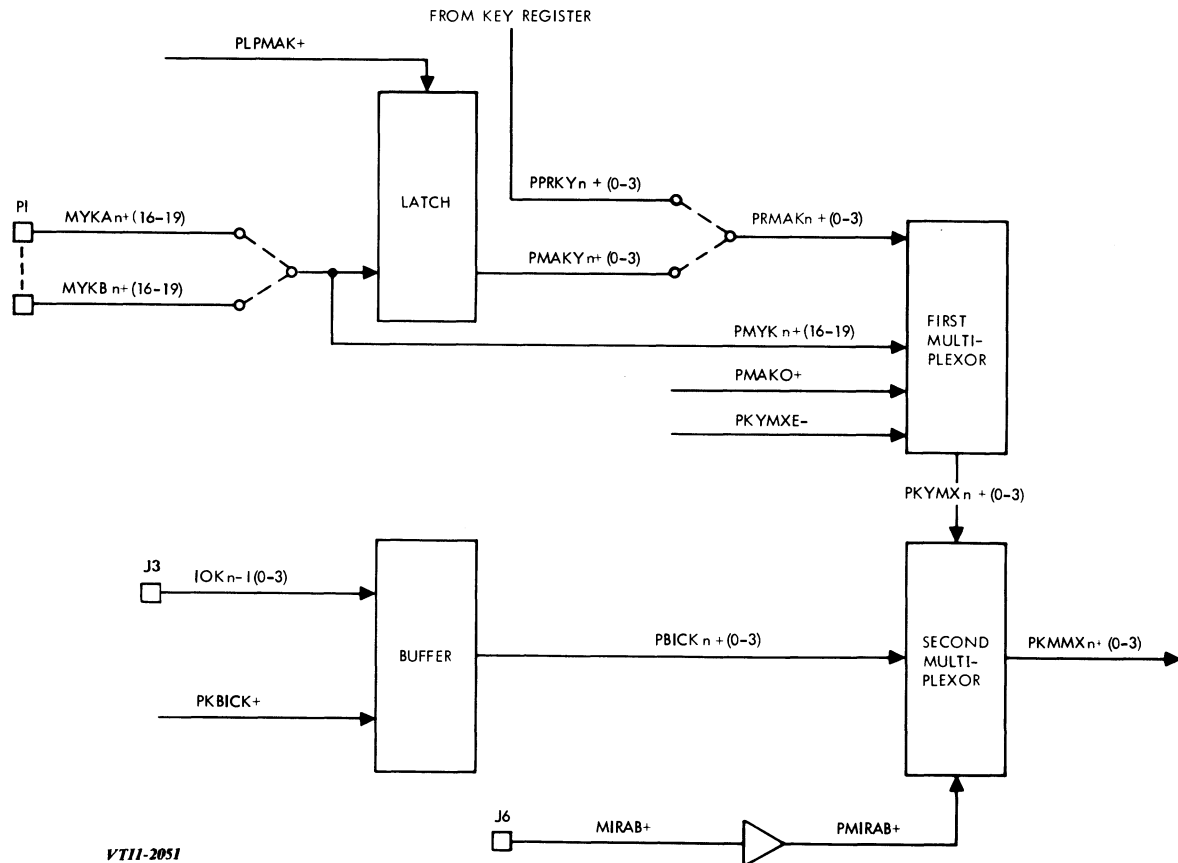
Input selection for the second multiplexor is controlled by I/O memory request signal $PMIRAB +$. A low $PMIRAB +$ transfers the first multiplexor output $PKYMX_n + (0-3)$ through the second multiplexor. A high $PMIRAB +$ selects the other input $PBICK_n + (0-3)$ from the buffer. BIC key bits $IOK_n - I(0-3)$ are transferred through the buffer on the positive-going transition of $PKBICK +$.

4.3.2 RAM Address Multiplexor

The RAM address multiplexor selects the RAM address consisting of $PRAM_n - (0-7)$ and $PKEYB_n - (0,1)$ from either the map-address counter or the combination of key and memory-address bits. Input selection for the multiplexor is controlled by the inhibit-memory-request signal $PIHMRQ -$. A low $PIHMRQ -$ selects contents of the map-address counter $PMAR_n + (0-9)$; a high $PIHMRQ -$ selects the key and memory-address bits $PKMMX_n + (0-3)$ and either $MYMA_n + (9-14)$ or $MYMB_n + (9-14)$.

4.3.3 Drivers and Row-Selection Decoder

The nine signal lines consisting of $PRAM_n - (0-7)$ and $PRAMW +$ are routed through 36 drivers to provide the power required for driving the RAM arrays.



VTII-2051

Figure 4-3. Key Multiplexor Block Diagram



THEORY OF OPERATION

The row-selection decoder consists of a selector that selects either $PMARn + (6,7)$ or $PKMMXn + (0,1)$ and a decoder that decodes the selector outputs. The decoded row-selection bits $PRAMSn - (0-3)$ select one of the four rows of the RAM arrays. Table 4-1 is a truth table for the row-selection decoder.

Table 4-1. Row-Selection Decoder Truth Table

INPUTS		ROW-SELECTION OUTPUTS			
$PMAR07-$ OR $PKMMX1+$	$PMAR06+$ OR $PKMMX0+$	$PRAMS3-$	$PRAMS1-$	$PRAMS2-$	$PRAMS0-$
L	L	L			
L	H		L		
H	L			L	
H	H				L

NOTE: L = low, H = high

VTII-2079

4.3.4 RAM Array

The RAM array is a 1024-word by 13-bit read/write memory that stores physical page addresses along with access- and swapping-control bits. As illustrated in figure 4-4, it consists of 256-word by 1-bit arrays arranged in 13

columns and 4 rows. Address bits $PRMAN - (0-7)$ (figure 4-5) select one of the 256 13-bit words in all four rows of the array. One of the four rows is selected by the associated row selector $PRAMSn -$ going low.

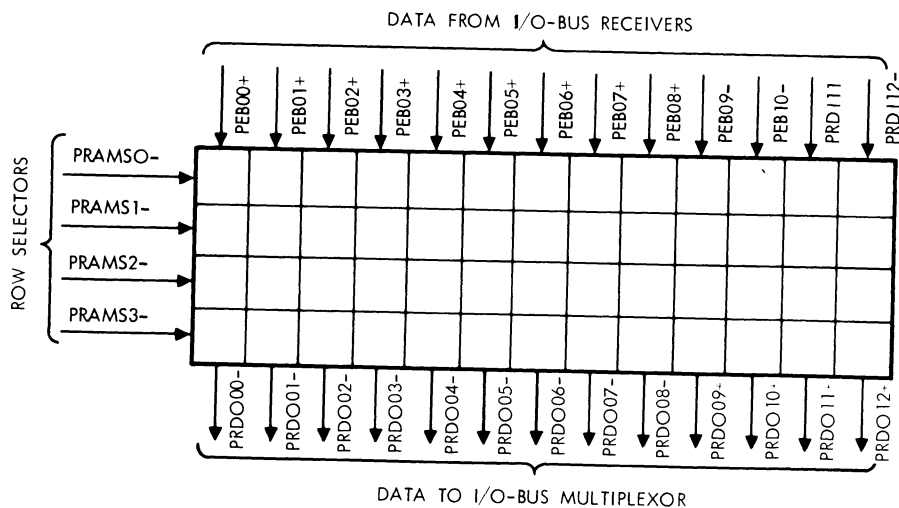
During a writing operation, a high $PRAMW +$ enables data from the I/O bus receivers to be loaded into the addressed location of the array. During a reading operation, a low $PRAMW +$ enables addressed data to be read out of the array.

4.3.5 I/O Bus Data Multiplexor

This multiplexor selects I/O-bus data from one of the following sources:

- RAM array: $PRDON - (0-8)$ and $PRDON + (9-12)$
- Instruction address register: $PIADn + (0-15)$
- Unassigned address register: $PUADn + (0-15)$
- Status signals: $PPRKYn + (0-3)$, $P64KEN +$, $PEMSnE + (0-3)$, $PDMAOE +$, $PDMAER +$, $PIOUER +$, $PIOEKn + (0-3)$
- DMA memory address register: $PDMA n + (0-15)$

As illustrated in figure 4-6, the I/O-bus data multiplexor consists of two multiplexors and a buffer. On positive transitions of the buffer-loading signal $PLDBF -$, RAM-array data are transferred through the buffer to one input of the second multiplexor. Under control of selector signals $PSIn + (0,1)$, the first multiplexor selects data from either the instruction-address register, unassigned address

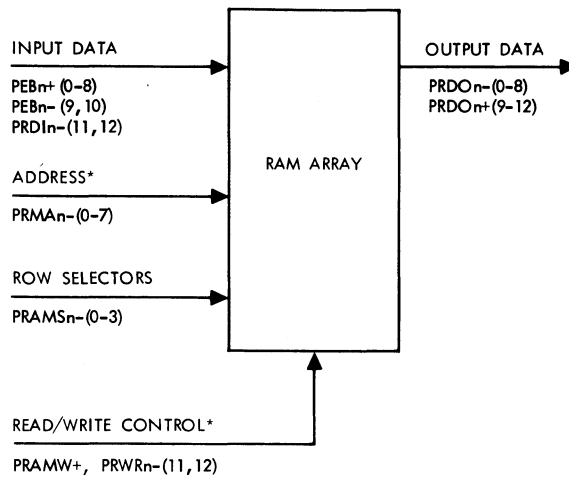


Notes:

- Each square represents 256 1-bit words.
- Read/write control $PRAMW +$ and address bits $PRMAN - (0-7)$ are not shown.

Figure 4-4. Layout of RAM Array

VTII-2052



*Address and read/write control signals are routed through drivers before they are applied to the array.

VT11-2053

Figure 4-5. RAM Array Block Diagram

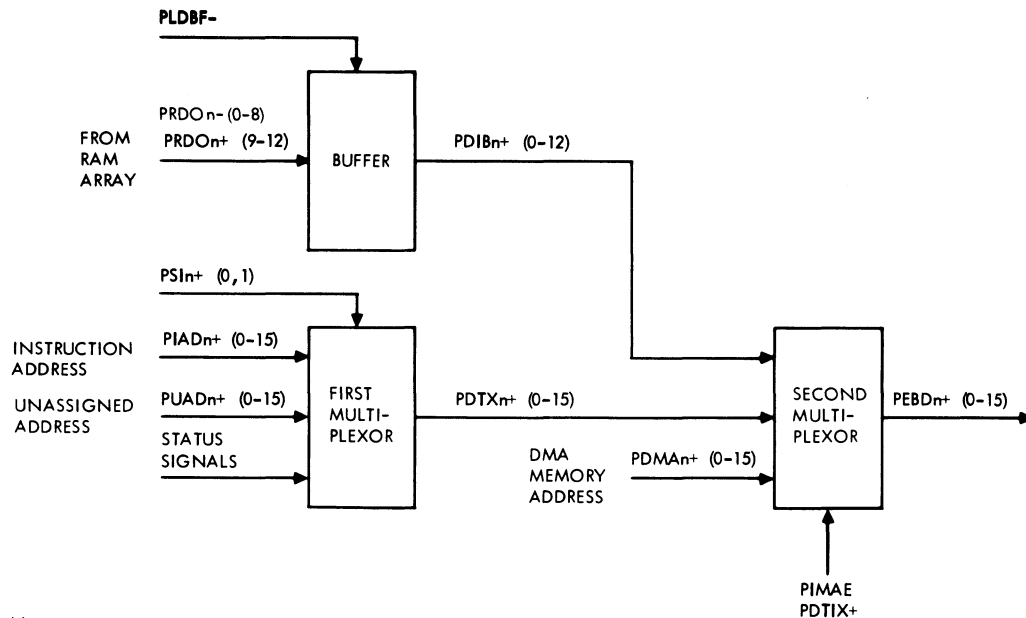
register, or internal-status register. Table 4-2 is a truth table for the first multiplexor. Under control of selector signals PIMAE- and PDTIX+, the second multiplexor selects data from either the buffer, first multiplexor, or DMA memory-address register. Table 4-3 is a truth table for the second multiplexor. Output data PEBDn+ (0-15) are applied to the I/O bus drivers.

4.3.6 I/O-Bus Drivers and Receivers

As illustrated in figure 4-7, the I/O-bus drivers transfer either a 16-bit data word EBnn- I(0-15) or a 4-bit interrupt address EBnn- I(1-4) onto the bidirectional I/O bus.

During an input-data transfer, a high PEBEN+ transfers a 16-bit data word through the drivers onto the I/O bus. When an interrupt address is being generated, a low PEBEN+ disables the sixteen drivers causing all of their outputs to go high (all zeros). This allows the four I/O-bus drivers to transfer the interrupt address onto the I/O bus. Table 4-4 lists the bit configurations for the various interrupt address generated by the memory map.

The I/O-bus receivers convert I/O-bus data into PEBn(0-15) for use in various circuits of the memory map.



Note:

Status signals consists of: PPRKYn+ (0-3), P64KEN+, PEMS nE+ (0-3), PDMAOE+, PDMAER+, PIOUER+, PIOEK n+ (0-3).

VT11-2054

Figure 4-6. I/O-Bus Data Multiplexor



THEORY OF OPERATION

Table 4-2. First Multiplexor Truth Table

SELECTORS		DATA INPUTS			OUTPUT
PS11-	PS10-	STATUS SIGNALS	PIADn+ (0-15)	PUADn+ (0-15)	PDTXn+ (0-15)
L	L	L			L
L	L	H			H
L	H		I		L
L	H		H		H
H	L			L	L
H	L			H	H
H	H				H

NOTE: L = low, H = high, no H or L = irrelevant input

VT11-2082

Table 4-3. Second Multiplexor Truth Table

SELECTORS		DATA INPUTS			OUTPUT
PDTIX-	PIMAE-	PDMA n+ (0-15)	PDIB n+ (0-12)	PDTX n+ (0-15)	PEBD n+ (0-15)
L	L	L			L
L	L	H			H
L	H		L		L
L	H		H		H
H	L			L	L
H	L			H	H
H	H				H

NOTE: L = low, H = high, no H or L = irrelevant input

VT11-2080

Table 4-4. Interrupt-Address Bit Configurations

I/O-Bus Bits EBn- I(1-4)				Interrupt Address (Octal)
4	3	2	1	
H	L	L	L	16
L	H	H	H	20
L	H	H	L	22
L	H	L	H	24
L	H	L	L	26
L	L	H	H	30
L	L	H	L	32
L	L	L	H	34

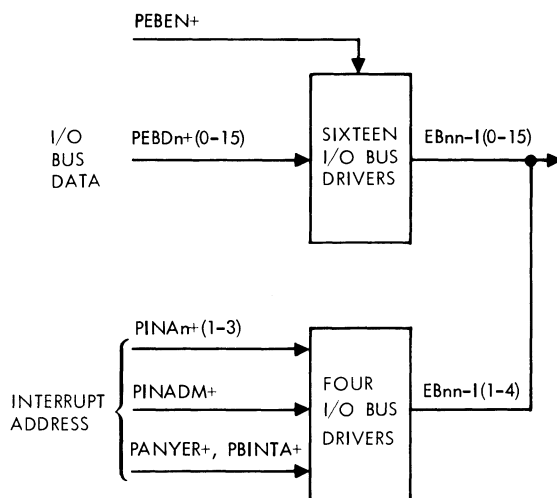
Note: L = low, H = high

4.3.7 Word-Transfer Counter

The word-transfer counter counts the number of words transferred during map loading and read-back operations. An 11-bit data word PEBn + (0-10), representing the number of words to be transferred, is loaded into the counter when PXFRW- is low. With each subsequent word-transfer, PDRYF + and PDTPF + go high causing the contents of the counter to decrease by one. When all words have been transferred, the output of the counter PWCZ- goes low. The low PWCZ- is sent to DMA control to indicate completion of the DMA transfers.

4.3.8 Key and 64K-Mode Registers

On the positive-going transition of the write-key control PWRKC +, PEBn + (0-3) and PEBO4 + are clocked into the key and 64K-mode registers, respectively. The 4-bit output of the key register PPRKYn + (0-3) is sent to the key multiplexor and to the I/O-bus data multiplexor as part of the status-signal input. The output bit of the 64K-mode register P64KEN + is also sent to the I/O-bus data multiplexor as part of the status-signal input. A high P64KEN + enables the 64K mode of operation. When P64KEN + is low, the 32K mode is used.



VT11-2055

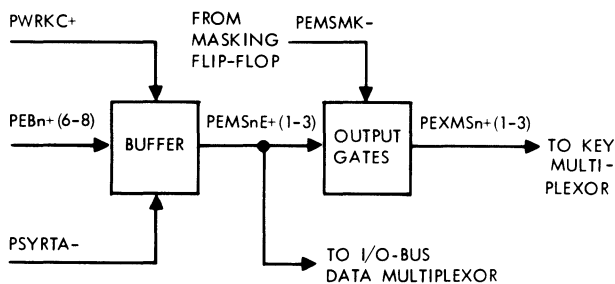
Figure 4-7. I/O Bus Drivers



4.3.9 Executive-State Register

This register (figure 4-8) stores one of the four states of the executive mode. On positive-going transitions of the write-key control $PWRKC+$, $PEBn+(6-8)$ are clocked into the executive-state buffer. This buffer is cleared with a low system reset $PSYRTA-$. Buffer-output bits $PEMSnE+(1-3)$ are applied to the I/O-bus data multiplexor as well as the executive-state output gates. When the executive-state masking flip-flop is reset, a high $PEMSMK-$ transfers the buffer-output bits through the output gates to the key multiplexor. Resetting of the flip-flop occurs during either the inactive mode of operation ($PACTV+$ low) or the decoding of an external control $EXC2\ 0546$ I/O instruction.

When an interrupt occurs, the executive-state masking flip-flop is set ($PEMSMK-$ low) causing all bits of $PEXMSn+(1-3)$ to go low. This results in executive-mode state 0 operation (executive map).



OUTPUT-BIT CONFIGURATION

PEXMSn+ 3	2	1	EXECUTIVE-MODE STATES
L	L	L	0
L	L	H	1
L	H	L	2
H	L	L	3

Note: L = low, H = high

VTII-2056

Figure 4-8. Executive-State Register Block Diagram

4.3.10 Map-Address Counter

When the I/O-write control $PWIOC-$ is low, the initial map address $PEBn+(0-9)$ is loaded into the counter and applied to the RAM address multiplexor. With each subsequent address-transfer, the content of the counter $PMARn+(0-9)$ is increased by one with the positive-going transition of $PIHMRQ-$. After the loading of the initial map address, the counter can count up to 1023 before resetting to zero.

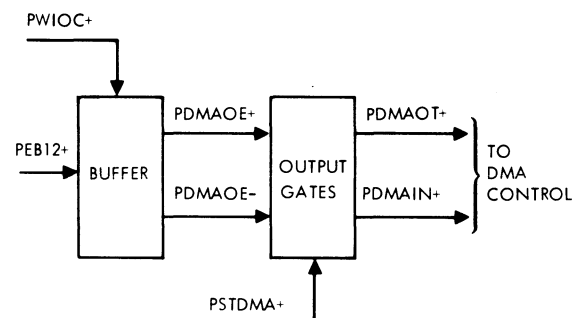
4.3.11 Input-Data Selection Register

On the positive-going transition of the I/O-write control $PWIOC+$, I/O data bits $PEBn+(10,11)$ are clocked into the input-data selection register. The register contents $PSIn+(0,1)$ are used to select one of three data inputs for the I/O-bus data multiplexor. Bit configurations for data-input selection are listed in the following table (L is low, H is high):

$PSI1+$	$PSI0+$	Selected Input Data
L	L	Status signals
L	H	Instruction address
H	L	Unassigned address

4.3.12 DMA I/O Register

On the positive-going transition of the I/O-write control $PWIOC+$, the I/O data bit $PEB12+$ is clocked into the DMA I/O register (figure 4-9). A low $PEB12+$ enables a DMA input transfer ($PDMAOE-$ high); a high $PEB12+$ enables a DMA output transfer ($PDMAOE+$ high). When the start-DMA-transfer I/O instruction ($EXC2\ 0346$) is decoded, a high DMA start signal $PSTDMA+$ transfers the DMA input or output signal ($PDMAIN+$ or $PDMAOT+$) through the output gates to the DMA control circuits. When the DMA-reset instruction ($EXC2\ 0446$) is decoded, the output gates are disabled with a low $PSTDMA+$.



VTII-2057

Figure 4-9. DMA I/O Register

4.3.13 DMA Memory-Address Counter

The DMA memory-address counter provides memory addresses for the processor during the address phase of map loading and read-back operations. When the loading signal $PWIMA-$ goes low, the initial memory address $PEBn+(0-15)$ is loaded into the counter and is then applied to the I/O-bus data multiplexor. With each subsequent address-transfer, the counter content $PDMAN+(0-15)$ is increased by one with the positive-going transition of $PIMAE-$. After the loading of the initial map address, the counter can count up to 65,535 before resetting to zero.



THEORY OF OPERATION

4.3.14 I/O-Error Key Register

When an error is detected during a DMA or PMA operation, the positive-going transition of $\text{PIOER}+$ clocks the error-key number $\text{PKMMXn}+(0-3)$ into the I/O-error key register. The register contents $\text{PIOEK}n+(0-3)$ are applied to the I/O-bus data multiplexor as part of the status-signal input.

4.3.15 Instruction Address Register

During preliminary decoding of each instruction, the instruction address register stores data from the processor's program counter. The register is updated with each instruction until an error is detected. Further updating is inhibited until the map's memory protection logic is enabled. During the error subroutine, the register contents are transferred to the processor by an input-data transfer instruction. The register contents equal $P+1$ where P is the logical address of the instruction that failed.

The inputs to the instruction address register $\text{PRAMAn-}(0-5)$ and $\text{PMYAn-}(0-8,15)$ come from the RAM address multiplexor and memory address receivers, respectively. On the positive-going transition of $\text{PLDIA}+$, these inputs are clocked into the register. During the time that the processor is performing preliminary instruction decoding ($\text{PCACID}+$ high) and the map's memory protection logic is enabled ($\text{PMPEN}+$ high), $\text{PLDIA}+$ goes high on the positive-going transition of the processor full clock $\text{PMFC}+$. The register contents $\text{PIADn}+(0-15)$ are applied to the I/O-bus data multiplexor. The instruction address is transferred to the processor by an input-data transfer instruction.

4.3.16 Unassigned Address Register

The same inputs that are applied to the instruction address register are also applied to the unassigned address register. When an unassigned address is detected, it is loaded into the unassigned address register on the positive-going transition of $\text{PUASER}+$. Clock signal $\text{PUASER}+$ is generated when output data bits 9 and 10 of the RAM array are both false, and the map's memory protection logic is enabled ($\text{PMPEN}+$ high). The register contents $\text{PUADn}+(0-15)$ are applied to the I/O-bus data multiplexor. The unassigned address is transferred to the processor by an input-data transfer instruction.

4.3.17 Swapping Control

Two swapping control bits are used by the memory map: change bit PDRI11- and usage bit (PRDI12-). The change bit indicates if a page has been written into since the bit was last reset. The usage bit indicates if a page has been

accessed since the bit was last reset. To facilitate scanning and counting the frequency of use, this bit is hardware reset whenever the processor reads the RAM array.

The swapping control circuit controls the setting and resetting of the swapping control bits and applies them to the RAM array.

During a map loading or read-back operation, a high $\text{PIHMRQ}+$ enables the state of PRDI11- to be the same as PEB11- . During a mapping operation, a low $\text{PIHMRQ}+$ causes PRDI11- to remain low regardless of the state of PEB11- .

During memory map loading ($\text{PDMAOT}+$ and $\text{PIHMRQ}+$ high), the state of PRDI12- is the inverse of $\text{PEB12}+$. If $\text{PDMAOT}+$ or $\text{PIHMRQ}+$ is low, the state of $\text{PEB12}+$ has no effect on PRDI12- .

4.3.18 Access Control

Access control bits $\text{PRDOn}+(9,10)$ are decoded to produce control signals for the four access control modes:

- Unassigned mode (PUNASG- low) indicates that the logical address is unassigned.
- Full-access mode (PFULAC- low) indicates that both reading and writing operations are permitted.
- Read-operand-only mode (PROPRD- low) indicates that only operand fetches are permitted. Instruction fetches from pages having this read-operand-only condition will not be executed.
- Read-only mode ($\text{PRDO9}+$ and $\text{PRDO10}+$ high) indicates that only instruction or operand fetches are permitted. Instruction fetches include both words of any double-word instruction as well as single-word instructions.

Table 4-5 is the decoder truth table.

Table 4-5. Access-Control Decoder Truth Table.

Inputs		Outputs		
$\text{PRDO10}+$	$\text{PRDO09}+$	PUNASG-	PFULAC-	PROPRD-
L	L	L	H	H
L	H	H	L	H
H	L	H	H	L
H	H	H	H	H

Note: L = low, H = high



4.3.19 Memory Address Drivers and Receivers

Seven bits from the RAM array $PRD0n-$ (0-6) and nine bits from the memory address receivers $PMYA n-$ (0-8) are applied to the memory address drivers. The driver outputs $MYA1n+$ (0-15), $MYA2n+$ (0-15), and $MYA3n+$ (0-15) are routed to the expansion memories.

Memory address bits 0 through 8 and bit 15 from either port A or B are applied to the memory address receivers. Receiver outputs are $PMYA n-$ (0-8,15).

4.3.20 Active/Inactive Register

This register provides active and inactive status of the memory map. In the executive mode, the register is set ($PACTV+$ high) indicating an active status. The register is set by the decoding of I/O instruction EXC2 0146 ($PAEXE+$ high) and a jump instruction ($PDJUMP+$ high).

In the inactive mode, the register is reset ($PACTV-$ high) indicating the inactive status. The register is reset by either a system reset ($PSYRTA-$ low) or by the decoding of I/O instruction EXC2 046 ($PINAE+$ high) and a jump instruction ($PDJUMP+$ high).

4.3.21 User/Executive Mode Register

This register provides control signals that indicate user- and executive-mode conditions. In the user mode, the

register is set by the decoding of I/O instruction EXC2 0246 ($PUSRE+$ high) and a jump instruction ($PDJUMP+$ high). The register output ($PUSER+$ high) is then gated with a high $PACTV+$ to produce the user-mode control ($PUSERM+$ high).

In the executive mode, the register is reset by the decoding of I/O instruction EXC2 0146 ($PAEXE+$ high) and a jump instruction ($PDJUMP+$ high). The register output ($PUSER-$ high) is then gated with a high $PACTV+$ to produce the executive-mode control ($PEXCM+$ high).

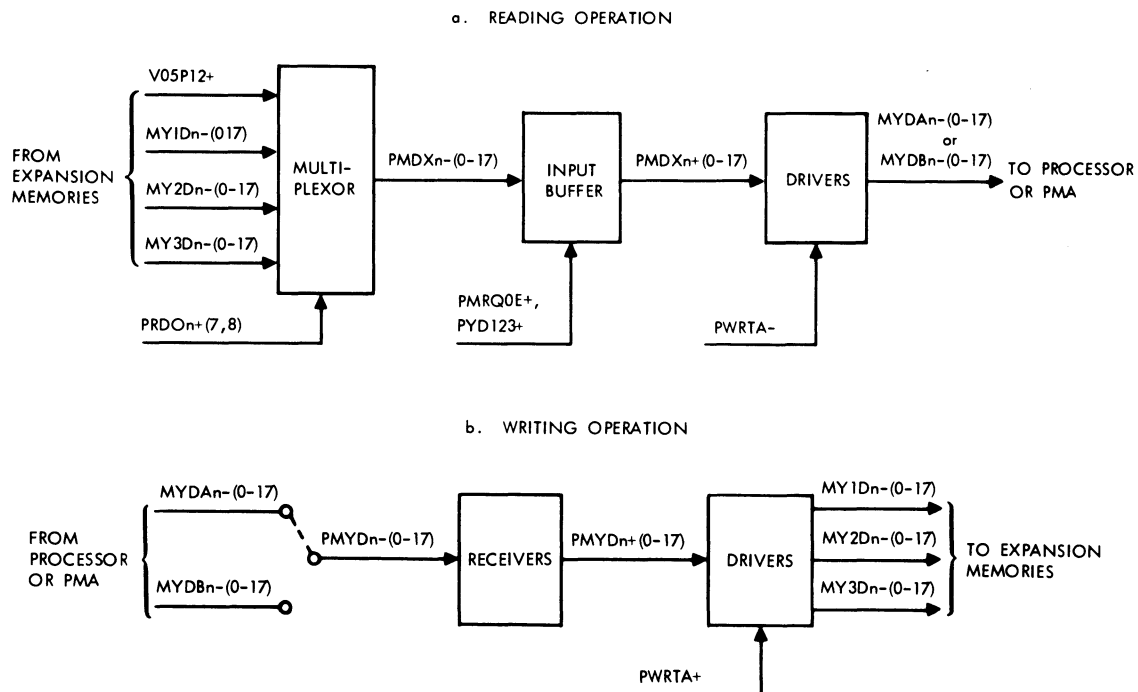
4.3.22 Bypass Drivers

When the memory map is in the inactive mode ($PACTV-$ high), address bits $PRAMAn-$ (0-5) and $PMYA15-$ bypass the RAM array and are transferred to memory port A or B. In addition, the high $PACTV-$ transfers memory request $MRQYA-$ or $MRQYB-$ to the mainframe memory.

When the memory map is active ($PACTV+$ high), data $PRD0n-$ (0-6) from the RAM array are transferred to memory port A or B.

4.3.23 Memory-Data Circuits

The memory data circuits consist of a multiplexor to select data from expansion memories, an input buffer to buffer the multiplexor output, and drivers and receivers to drive and receive memory data. Figure 4-10 shows the data paths through the memory map between the processor and expansion memories for reading and writing operations.



VT11-2058

Figure 4-10. Memory-Data Paths Between Processor and Expansion Memories



THEORY OF OPERATION

During a reading operation (figure 4-10a), data from the expansion-chassis memories are applied to the multiplexor. Data selection is controlled by PRD07+ and PRD08+ from the RAM array (refer to table 4-6 for truth table). When data from the mainframe memory are read, both PRD07+ and PRD08+ are low causing no data to be transferred through the multiplexor. When enabled by either mainframe memory request (PMRQOE+ high) or a memory acknowledgment from expansion chassis 1, 2, or 3 (PYD123+ high), the input buffer inverts the multiplexor output PMDXn- (0-17) and applies it to the drivers. A high PWRTA- causes the drivers to invert the buffer output PMDXn+ (0-17) and transfer it onto the memory bus. If data from the mainframe memory are read, all driver output bits are high. This occurs because mainframe memory data are transferred to the processor without going through the memory map.

Table 4-6. Memory-Data Multiplexor Truth Table

Selectors		Data Inputs				Output
PRD08 +	PRD07 +	V05P12 +	MY1Dn- (0-17)	MY2Dn- (0-17)	MY3Dn- (0-17)	PMDXn- (0-17)
L	L	H				H
L	H		L			L
L	H		H			H
H	L			L		L
H	L			H		H
H	H				L	L
H	H				H	H

Notes: 1. L = low, H = high, no H or L = irrelevant input
2. V05P12+ is a constant high voltage level

During a writing operation (figure 4-10b), memory data from memory port A or B are applied to receivers. The receivers invert the data and apply them to drivers. A high PWRTA+ causes the drivers to invert the receiver output PMYDn+ (0-17) and transfer it onto the three expansion memory buses. The memory request MRQY1+, MRQY2+, or MRQY3+ generated by the memory control circuit determines which expansion memory accepts the memory data.

4.3.24 Address Detection and Function Control

Address signal PADR46+ goes high when address 046 is decoded (or alternate address 056) from bits PEBn(0-5) and there are no DMA or interrupt requests (IUAX- I high). An external control (EXC2) instruction is decoded when PEB15+ is high and PADR46+ is gated with the function ready control FRYX- I (PFRY46+ high). As illustrated in table 4-7, specific EXC2 instructions are executed by decoding the function code bits PEBn+(6,7,8). The decoded outputs correspond to the various EXC2 instructions and are described as follows:

- PINAEX- places memory map in the inactive mode (EXC2 046).

Table 4-7. EXC2 Decoder Truth Table

INPUTS			OUTPUTS						
PEB08-	PEB07-	PEB06-	PINAEX-	PACTEX-	PUSREX-	PSDMEX-	PRDMEX-	PCLMEX-	PEMPLEX-
L	L	L	L						
L	L	H		L					
L	H	L			L				
L	H	H				L			
H	L	L					L		
H	L	H						L	
H	H	L							L
H	H	H							L

Notes:

- L = low, H = high
- For clarity, only the low (or true) states of the outputs are included in the table.

VTII-2081

- PACTEX- switches memory map from inactive mode to executive mode (EXC2 0146).
- PUSREX- switches memory map from executive mode to user mode (EXC2 0246).
- PSDMEX- starts a memory-map DMA transfer (EXC2 0346).
- PRDMEX- resets the memory map's DMA-transfer logic (EXC2 0446).
- PCLMEX- removes the executive-mode mask (EXC2 0546).
- PEMPLEX- enables the memory protection function and instruction address updating (EXC2 0646).
- PDMPEX- disables the memory protection function and instruction address updating (EXC2 0746).

Function code bits PEBn+(6,7,8) are also decoded for sense (SEN) instructions (PEB12+ high). Table 4-8 lists the function codes for the SEN instructions. If the condition sensed by the instruction is true, a sense response (SERX- I low) is sent to the processor.

Table 4-8. Function Codes for SEN Instructions

PEB08 +	PEB07 +	PEB06 +	Instruction
L	L	L	SEN 046 (SEN0+ high)
L	L	H	SEN 0147 (SEN1+ high)

Note: L = low, H = high

Execution of input- and output-data transfer instructions consists of an instruction phase and a data phase. A high PEB13+ indicates an input-data transfer and high PEB14+ indicates an output-data transfer. When gated with a high PFRY46+, these signals start the instruction phase by setting the input- or output-transfer flip-flop (PDTIX+ or PDTOX+ high). The data phase is initiated when a low data-ready signal DRYX- resets the input- or output-transfer flip-flop. Control signals for the three



THEORY OF OPERATION

formats of output-data transfers are decoded from PEB14 + and PEB15 + (table 4-9). The control signals are:

- PWRKC⁻, write key control
- PWIOC⁻, write I/O control
- PXFRW⁻, write number of words to be transferred via DMA

Table 4-9. Output-Transfer Format Truth Table

Inputs		Outputs		
PEB15 +	PEB14 +	PWRKC ⁻	PWIOC ⁻	PXFRW ⁻
L	L	H	H	H
L	H	L	H	H
H	L	H	L	H
H	H	H	H	L

Note: L = low, H = high

4.3.25 Memory Control

By decoding output bits 7 and 8 of the RAM array with various other control signals, the memory control applies one of the memory request signals MRQYA⁻, MRQYB⁻, or MRQYn + (1-3) onto the appropriate memory bus (mainframe or an expansion memory).

In addition, the memory control provides the following control signals:

- Read/write control signals for left and right memory bytes are applied to selected memory bus.
- System reset (SRST⁻) and power-failure reset (SPFA⁻) signals are applied to the selected memory bus.
- Memory lockout signals MHGYn⁻ (1-3) or MHMYn⁻ (1-3) are applied to the selected expansion-memory bus.
- The expansion-memory acknowledgement signals YDNMn + (1-3) are delayed 30 nanoseconds and is transferred to the mainframe memory (YDNMA + or YDNMB +). The delay ensures that data on the mainframe memory bus has stabilized.

4.3.26 DMA Control

The DMA control controls the memory-map loading and read-back operations. The following events occur during the loading operation (for timing waveforms see figure 4-14):

- The processor executes the EXC2 0346 instruction to start the memory-map DMA transfer.
- The memory map generates a trap-out request (TPOF⁻ low).

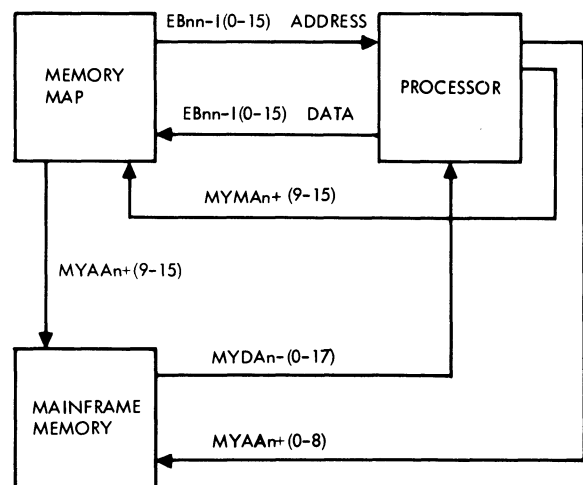
c. The processor transfers the address to the memory and the data to the memory map via the I/O bus.

d. The memory map generates a write strobe (PRAMW + high) to load the data into the RAM array.

The following events occur during the read-back operation (for timing waveforms refer to figure 4-15):

- The processor executes the EXC2 0346 instruction to start the memory-map DMA transfer.
- The memory map loads read-back data PRDOn(0-12) into buffer PDIBn + (0-12).
- The memory map resets the usage bit (PRDI12⁻).
- The memory map generates a trap-in request (TPIF⁻ low).
- When the processor acknowledges the trap-in request, the memory map places the read-back data onto the I/O bus for transferal to the processor.

The data flow for the map loading operation with port A of the mainframe memory is shown in figure 4-11. The data flow begins with the memory map sending a logical address EBnn-I(0-15) to the processor via the I/O bus (this occurs during the PFRYF + phase of the DMA transfer). The processor sends the address to the mainframe memory mapped or unmapped depending on whether the memory map is active or inactive. Data contained at this physical address are then transferred to the processor as MPDAn-(0-17). To complete the loading operation, the processor then loads the data into the memory map via the I/O bus.



VTII-2059

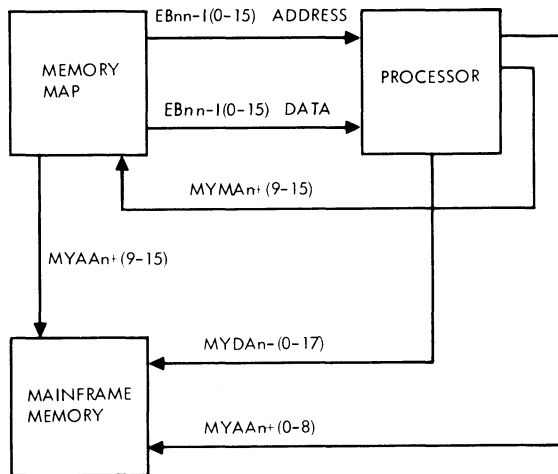
Figure 4-11. Data Flow for Map Loading Operation



THEORY OF OPERATION

The data flow for the map read-back operation with port A of the mainframe memory is shown in figure 4-12. As in the loading operation, the data flow begins with the memory map sending a logical address $EBnn-I(0-15)$ to the processor via the I/O bus (during the $PFRYF+$ phase of the DMA transfer). During the $PDRYF+$ phase, the memory map transfers RAM-array data onto the I/O bus and the processor transfers this data to the memory data bus. As in the loading operation, the processor applies the memory address to the mainframe memory and memory map. Data on the memory data bus are then written into the mapped physical address, thus completing the read-back operation.

Timing waveforms for the memory-map loading and read-back operations are shown in figures 4-14 and 4-15.



VTII-2060

Figure 4-12. Data Flow for Map Read-Back Operation

4.3.27 Mode Switching Control

This section controls switching of the memory-map modes of operation. As described in section 3.4, the modes of operation can be switched as follows:

- Executive mode to inactive mode
- Inactive mode to executive mode
- Executive mode to user mode

- User mode to executive mode

Timing waveforms for the mode switching operations are shown in figures 4-24 through 4-27.

4.3.28 Interrupt Control

This section provides interrupt control signals for the memory-protection and DMA-completion interrupts.

The following events occur with a memory-protection interrupt (for timing waveforms see figure 4-23):

- The memory map detects an error ($PANYER+$ high).
- The memory map synchronizes with the interrupt clock ($PIUCX+$), and raises an interrupt request ($PBINTe+$ high) which in turn sets the I/O bus interrupt request ($IURX-I$ low).
- Upon receipt of the interrupt acknowledgment ($IUAX-I$ low), the memory map places the interrupt address on the I/O bus until the trailing edge of $PFRYX+$.
- On the negative-going transition of $PIUCX+$, the memory map resets the interrupt request ($IURX-I$ high).

The following events occur with a DMA-completion interrupt (for timing waveforms see figure 4-16):

- A DMA transfer operation is complete when either the word-transfer counter is set to zero ($PWCZ-$ low) or an error termination is detected ($PDMTRM+$ high).
- The memory map synchronizes with the interrupt clock ($PIUCX+$) and sets an interrupt request ($PIURF+$ high) which in turn sets the I/O bus interrupt request ($IURX-I$ low).
- From this point on, the operation is the same as in steps c and d of the memory-protection interrupt.

4.4 MEMORY MAP TIMING

This section provides timing waveforms for the various memory-map operations. The operations with their figure numbers are listed below:

Figure 4-13, Programmed I/O Data Transfer

Figure 4-14, Memory-Map Loading via High-Speed DMA

Figure 4-15, Memory-Map Read-Back via High-Speed DMA

Figure 4-16, Memory-Map Loading/Read-Back Termination

Figure 4-17, Memory Mapping

Figure 4-18, I/O and Halt Error Detection

Figure 4-19, Jump-Error Detection

Figure 4-20, Unassigned and Writing Error Detection

Figure 4-21, Instruction-Fetch Error Detection

Figure 4-22, I/O Data-Transfer Error Detection

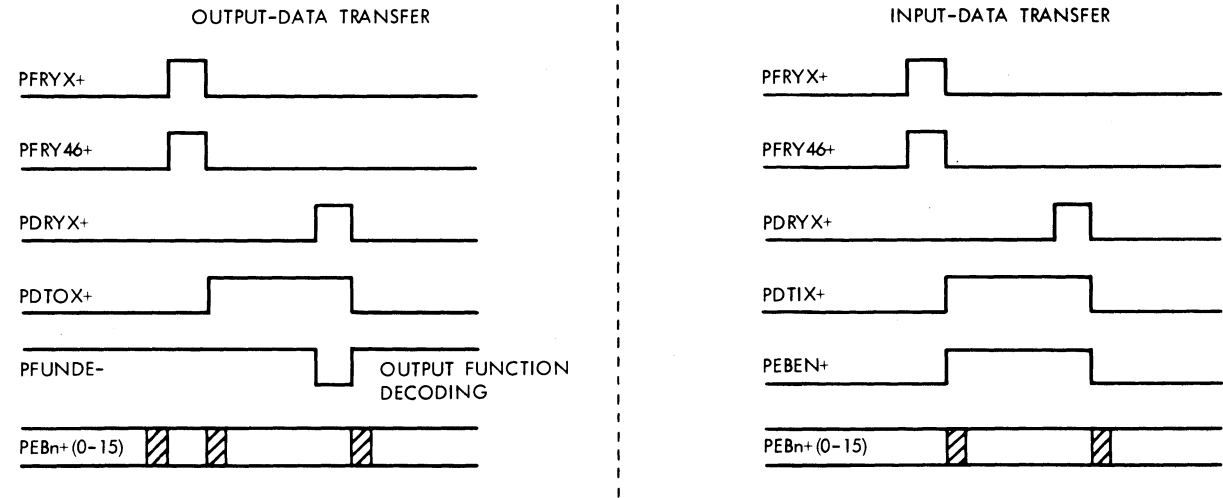
Figure 4-23, Memory Protection Interrupt

Figure 4-24, Executive-Mode to Inactive-Mode Switching

Figure 4-25, Inactive-Mode to Executive-Mode Switching

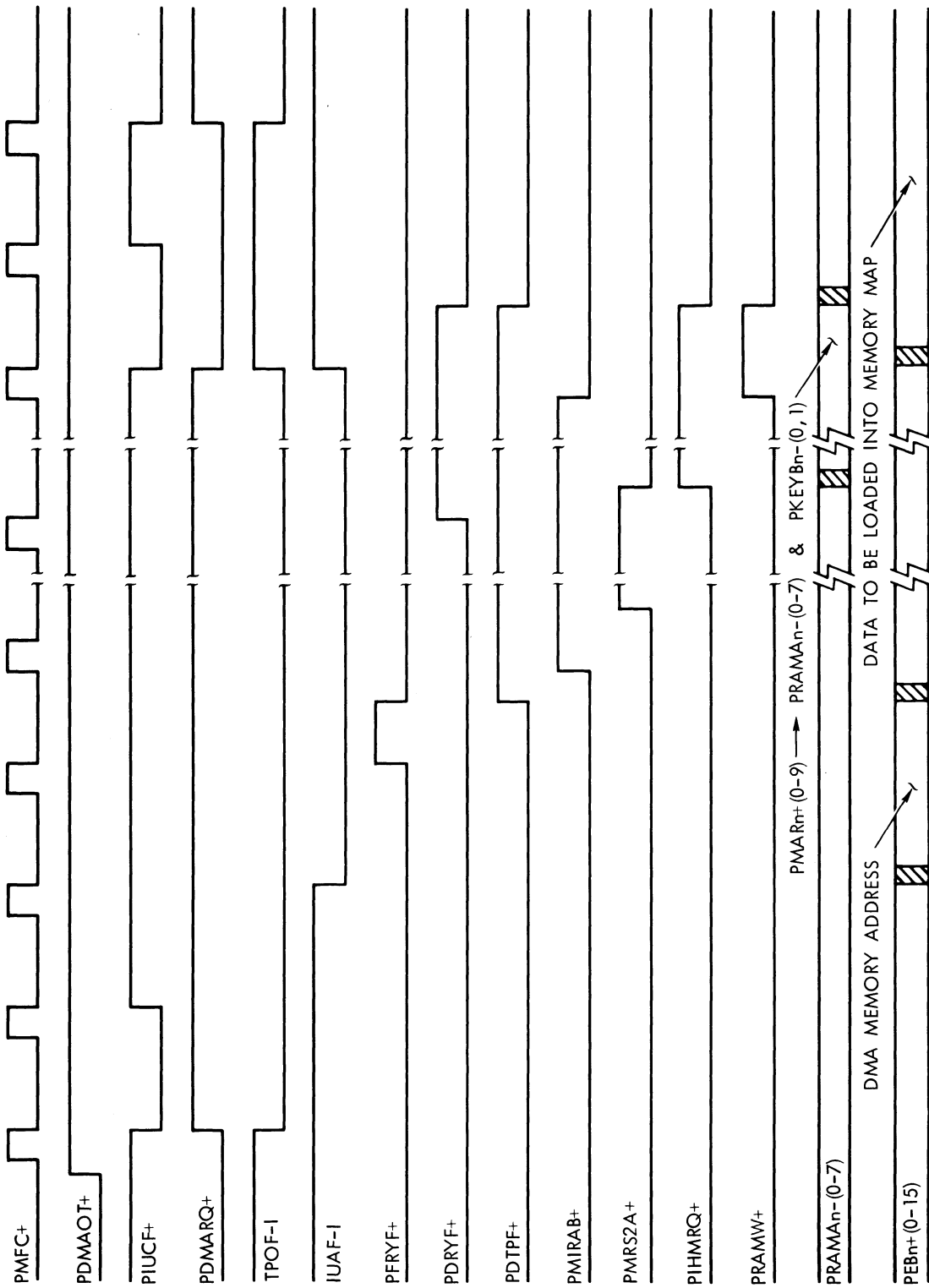
Figure 4-26, Executive-Mode to User-Mode Switching

Figure 4-27, User-Mode to Executive-Mode Switching



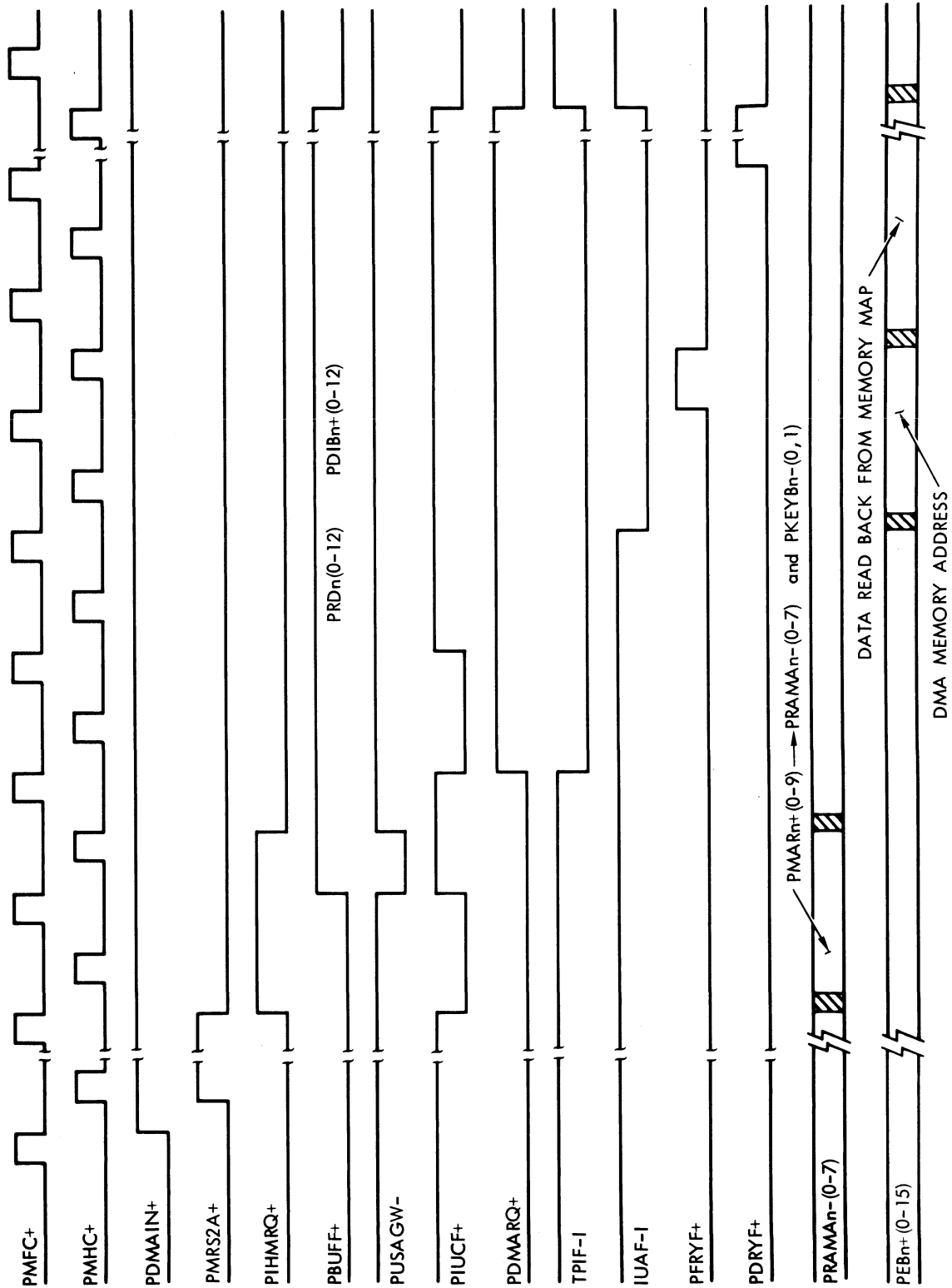
VTII-2061

Figure 4-13. Programmed I/O Data Transfer



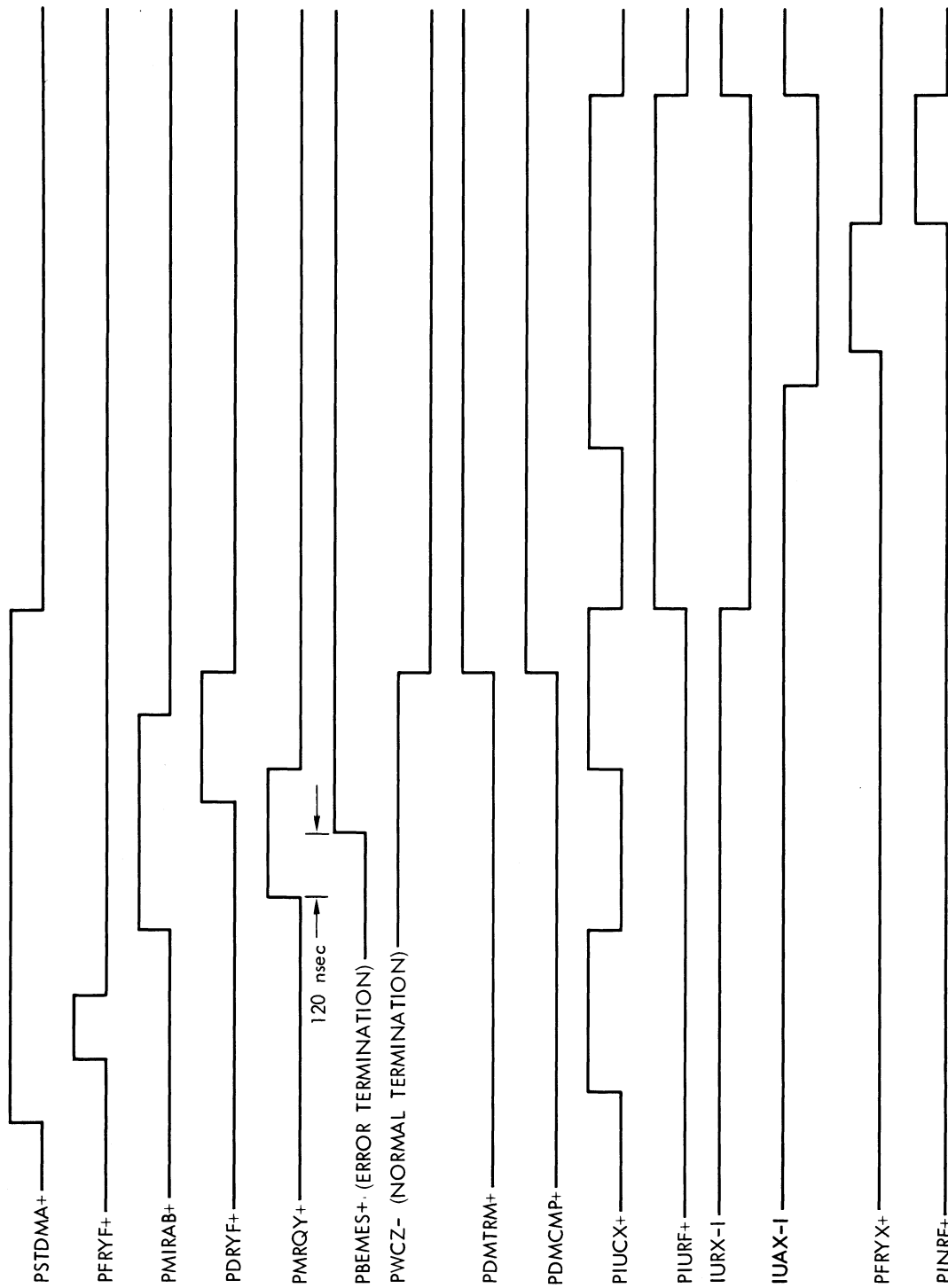
VTII-2062

Figure 4-14. Memory-Map Loading via High-Speed DMA



VTII-2063

Figure 4-15. Memory-Map Read-Back via High-Speed DMA



VTII-2064

Figure 4-16. Memory Map Loading/Read-Back Termination

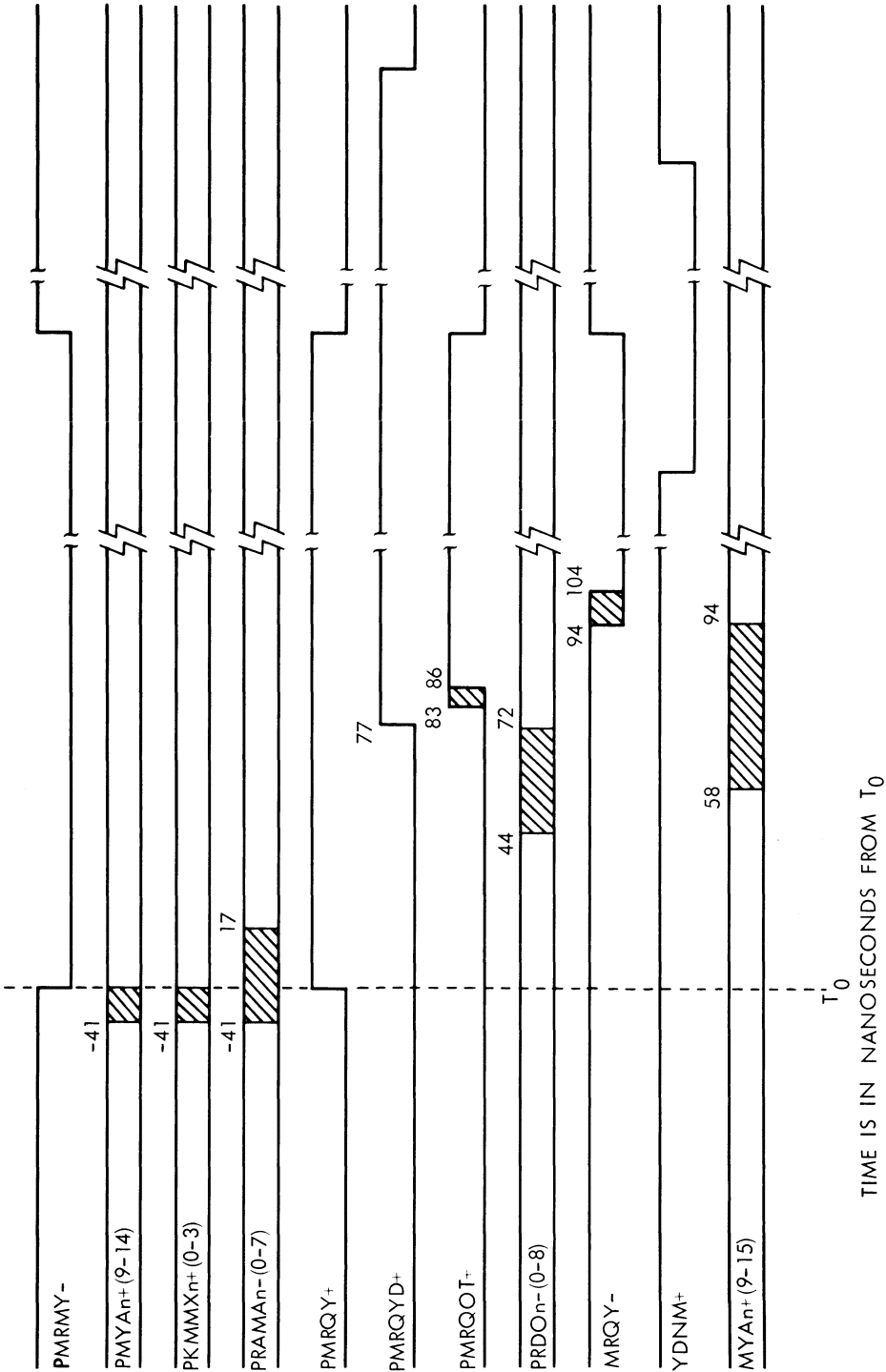
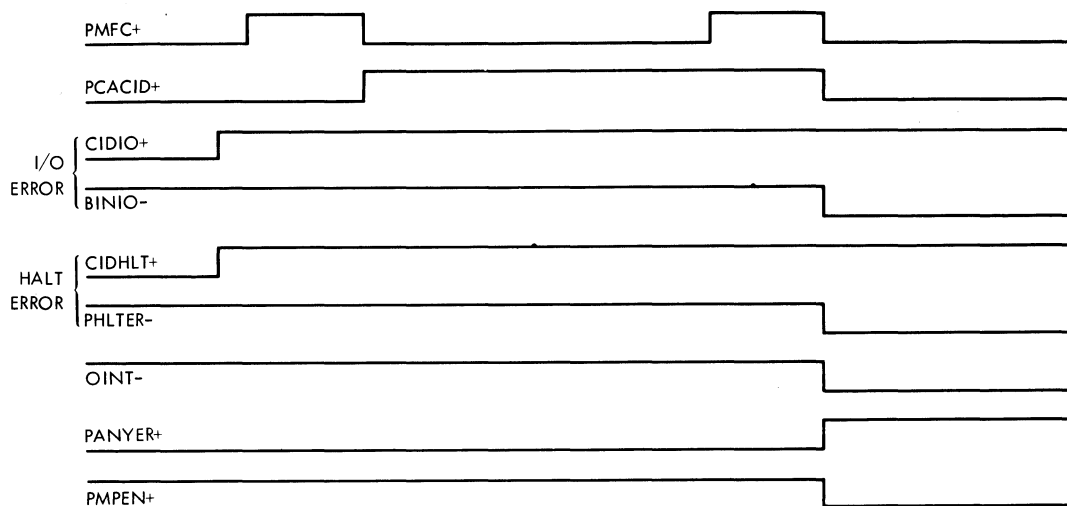


Figure 4-17. Memory Mapping

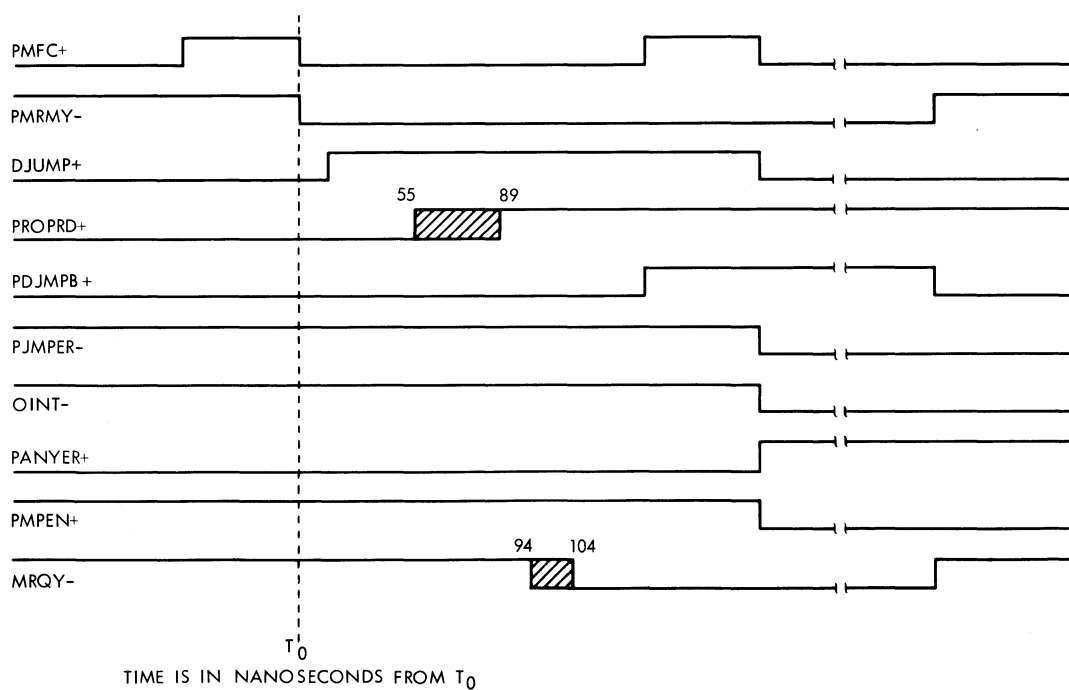


THEORY OF OPERATION



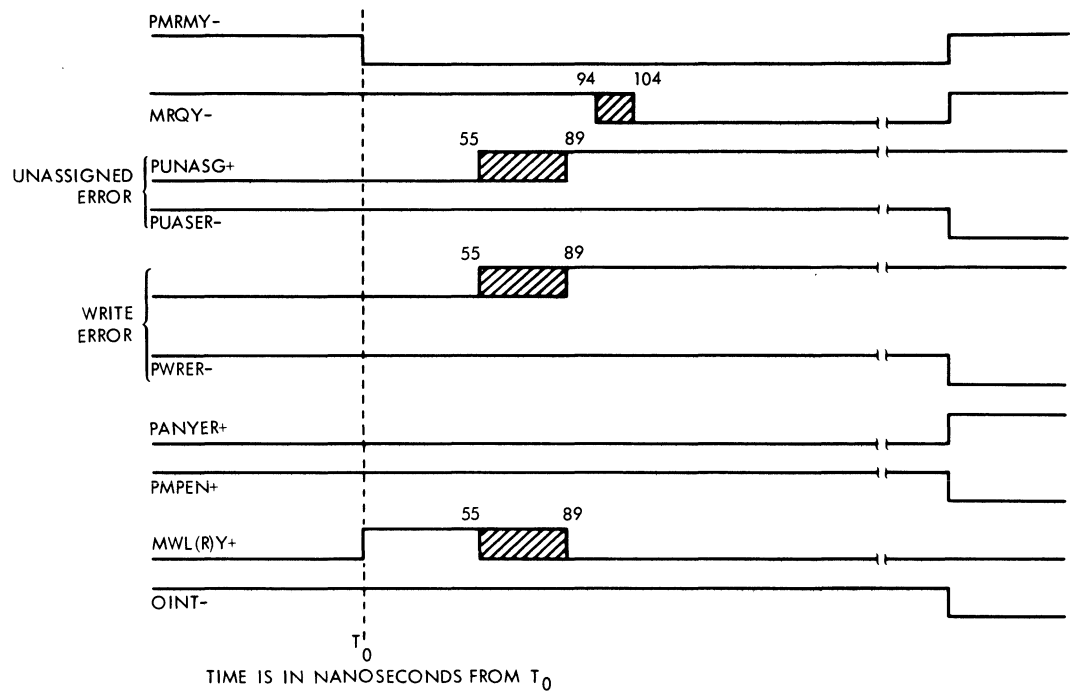
VTII-2066

Figure 4-18. I/O and Halt Error Detection



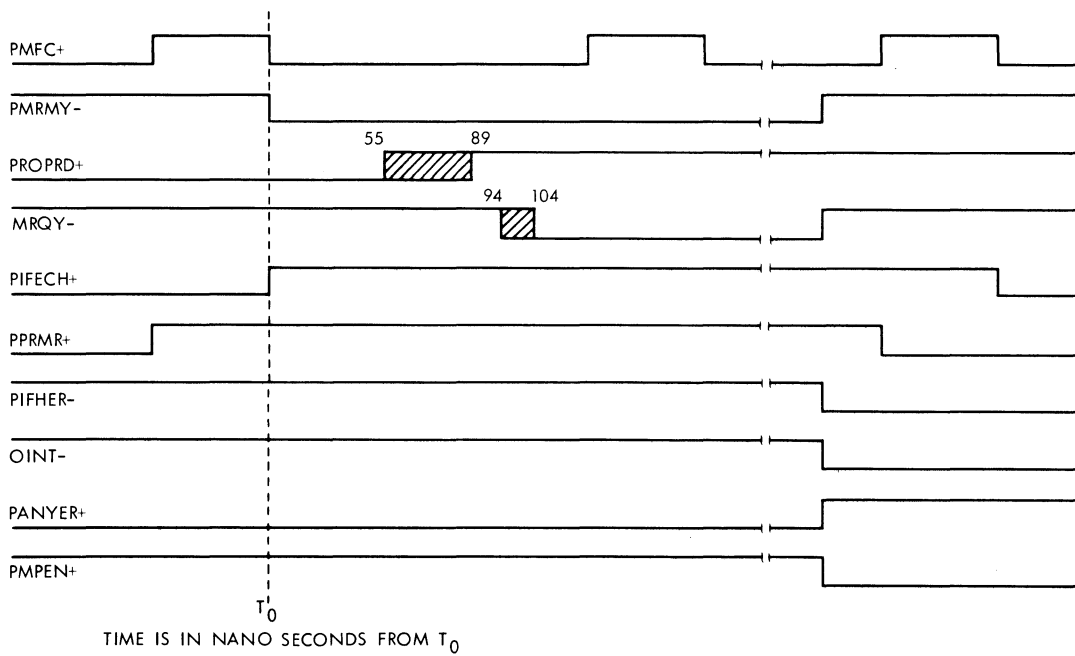
VTII-2067

Figure 4-19. Jump-Error Detection



VT11-2068

Figure 4-20. Unassigned and Writing Error Detection

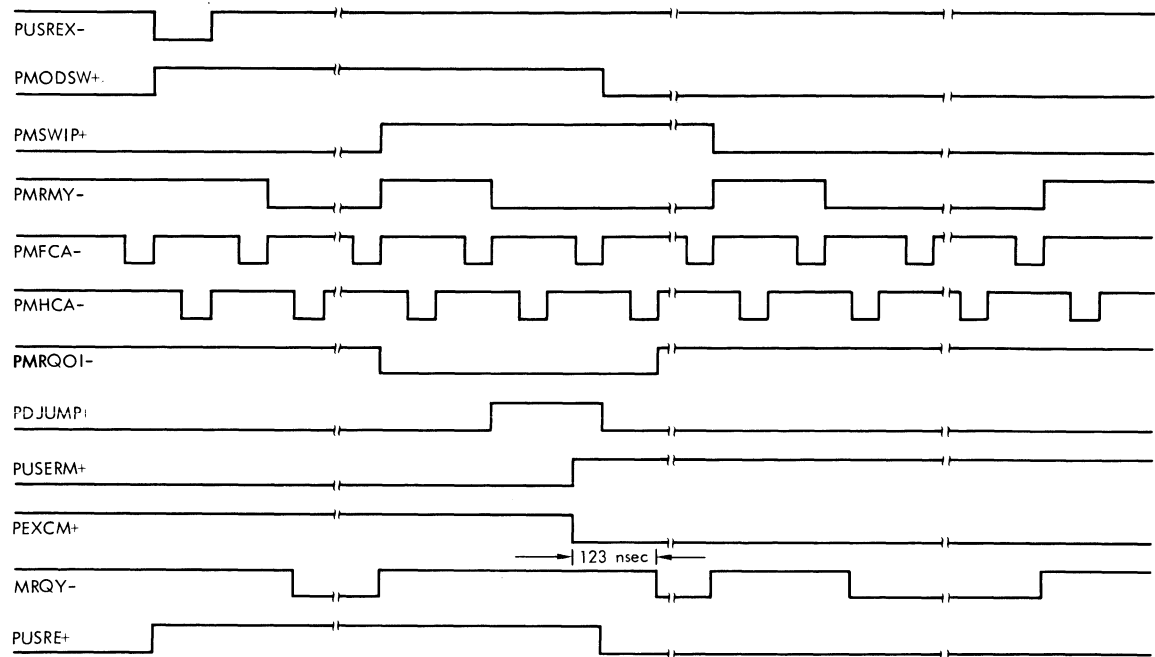


VT11-2069

Figure 4-21. Instruction-Fetch Error Detection

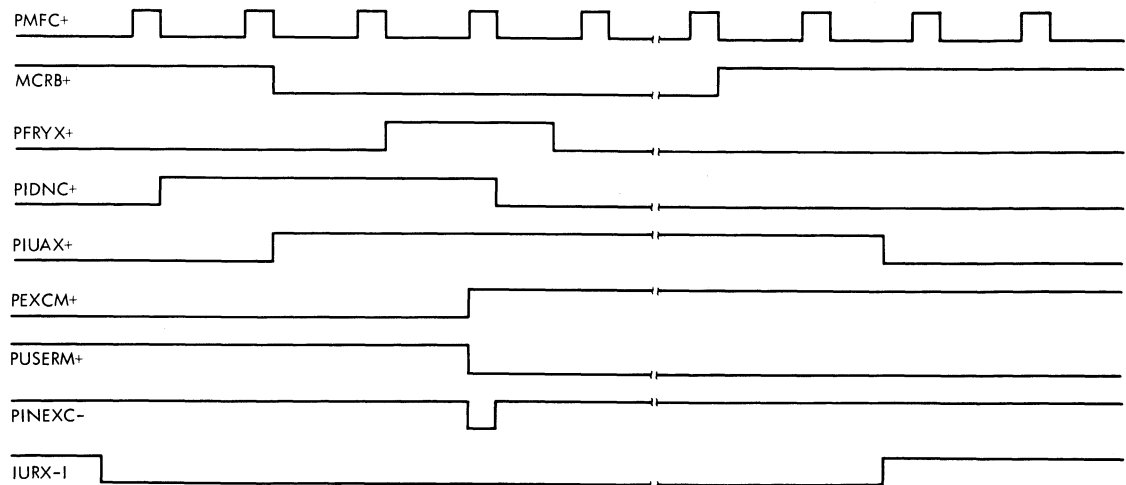


THEORY OF OPERATION



VT11-2074

Figure 4-26. Executive-Mode to User-Mode Switching



VT11-2075

Figure 4-27. User-Mode to Executive-Mode Switching



SECTION 5

MAINTENANCE

Maintenance personnel should be familiar with the contents of this manual before attempting memory map troubleshooting. A test program (section 7) is available to verify correct operation and to isolate malfunctions to a particular section of the memory map. Further diagnosis can then be made by referring to this manual.

5.1 TEST EQUIPMENT

The following test equipment and tools are recommended for memory map maintenance:

- a. Oscilloscope, Tektronix type 547 with dual-trace plug-in unit or equivalent.
- b. Multimeter, Triplett type 630 or equivalent.
- c. Soldering iron, 15-watt pencil type.

5.2 CIRCUIT BOARD REPAIR

The memory map board is a four-layer PC board. The two outer layers provide signal interconnections for the circuit components. The two inner layers provide low-impedance ground and power-voltage distribution, and 90-ohm micro-strip transmission lines for all signals. The ICs contained

on the board consist of LSI memories; MSI multiplexors, decoders, and registers; and SSI gates and flip-flops.

If it has been determined that circuit board repair is required, it is recommended that the Varian Data Machines customer service department be contacted so that a new circuit board can be installed in the user's system and the faulty one returned to the factory for repairs. However, if the user decides to perform his own repairs, extreme caution should be used so that the circuit board is not permanently damaged. Approved repair procedures should be followed such as the ones described in document IPC-R-700A prepared by the Institute of Printed Circuits.

5.3 CIRCUIT-COMPONENT IDENTIFICATION

For IC components, the memory map board has location coordinates that are used in the logic diagrams as reference designations. For example, a flip-flop designated C8 in the memory map logic diagram is in the IC package at location row C column 8 on the memory map board. For discrete components, the reference designations used in the logic diagrams appear on the circuit board adjacent to each component.

Parts lists in volume 2 provide a cross reference between Varian and the manufacturers part numbers.



varian data machines



SECTION 6

MNEMONICS

This section presents an alphabetized list of memory map signal mnemonics with definitions.

Plus or minus signs are included at the end of each mnemonic. The plus sign indicates the signal is at a high logical level when its function is being performed. The minus sign indicates the signal is at a low logical level when its function is being performed. A signal that is the logical inversion of another uses the same mnemonic with an opposite sign; these signals are complements of each other.

I/O bus signal mnemonics end with - I.

Mnemonic	Description	Mnemonic	Description
AREAD-	Read/write signal from the PMA option. A low level indicates a PMA reading operation.	DRYX- I	Data ready for normal DMA operation.
BIMES- I	Stops a DMA transfer due to an error during a memory-mapping operation.	EBnn- I(0-15)	I/O-bus data.
BINIO-	Indicates an I/O instruction error has occurred.	FRYF- I	Function ready for high-speed DMA operation.
BINTE-	Memory-protection interrupt priority.	FRYX- I	Function ready for normal DMA operation.
BTMES- I	Stops a PMA transfer due to an error during a memory-mapping operation.	IDNC-	I/O done signal from option board.
CACIDE +	Used by the processor to transfer instruction-decoder contents onto the control-store address bus.	IOKn- I(1-4)	I/O key bits from BIC.
CIDHLT +	Indicates that the processor has decoded a halt instruction.	IUAF- I	Interrupt acknowledgment for high-speed DMA operation.
CIDIO +	Indicates that the processor has decoded an I/O instruction.	IUAX- I	Interrupt acknowledgment for normal DMA operation.
CIDJMK +	Indicates that the processor has decoded a jump-and-mark instruction.	IUCF- I	Interrupt clock for high-speed DMA operations.
DJUMP +	Indicates that the program has been directed to the effective jump address of a jump instruction.	IUCX- I	Interrupt clock for normal DMA operation.
DRYF- I	Data ready for high-speed DMA operation.	IURX- I	I/O-bus interrupt request.
		IWLMC-	I/O-write left byte.
		IWRMC-	I/O-write right byte.
		MAKO +	From processor, indicating the PMA memory request has been acknowledged.
		MFC-	Processor full clock.
		MHC-	Processor half clock.
		MHGY-	Inhibits all memory access on port A of the mainframe memory.
		MHGYn- (1-3)	Inhibits all memory access on port A of the expansion memories.
		MHMY-	Inhibits all memory access on port B of the mainframe memory.



MNEMONICS

Mnemonic	Description	Mnemonic
MHMYn- (1-3)	Inhibits all memory access on port B of the expansion memories.	MYMBn + (9-15)
MIMCn + (0,1)	Contains bits 0 and 1 of the IM field to specify a reading or writing operation.	MY1Dn- (0-17)
MIRAB +	I/O memory request.	MY2Dn- (0-17)
MRMYA-	Memory-map memory request of port A.	MY3Dn- (0-17)
MRMYB-	Memory-map memory request of port B.	OINT-
MRQY-	Mainframe memory request.	P64KEN +
MRQYn- (1-3)	Expansion memory requests.	PACTEX-
MRSZA-	Memory sequencing flip-flop in processor.	PACTV +
MWLY +	Mainframe-memory write, left byte.	PADR46 +
MWLYn + (1-3)	Expansion-memory write, left byte.	PADSEL +
MWRY +	Mainframe-memory write, right byte.	PANYER +
MWRYn + (1-3)	Expansion-memory write, right byte.	PBINTE +
MYAn + (0-15)	Mainframe memory address bits.	PBEMES +
MYA1n + (0-15)	Address bits for expansion memory 1.	PBICKn + (0-3)
MYA2n + (0-15)	Address bits for expansion memory 2.	PBINIO +
MYA3n + (0-15)	Address bits for expansion memory 3.	PBINRS +
MYDAn- (0-17)	Mainframe memory data, port A.	PBINTA +
MYDBn- (0-17)	Mainframe memory data, port B.	PBUFE-
MYKAn + (16-19)	PMA key bits, port A.	PBUFF +
MYKBn + (16-19)	PMA key bits, port B.	PCLMEX-
MYMAN + (9-15)	Map memory address bits, port A.	PDIBn + (0-12)



MNEMONICS

Mnemonic	Description
PFRYAR +	Indicates that a function ready condition has occurred when IUA is true.
PFULAC +	Full-access mode.
PFUNDE-	Enabling signal for function decoding.
PHLTER-	Halt-instruction error.
PIADn + (0-15)	Output of instruction register.
PIFECH +	Instruction fetch.
PIFERE +	Instruction-fetch error.
PIFHER +	Instruction fetch error.
PIFJMK-	Jump-and-mark instruction fetch.
PIHMRQ +	Inhibits memory request.
PIMAE-	Memory address error for loading and read operation.
PINAn + (1-3)	Interrupt-address data.
PINADM +	Interrupt address data.
PINAE +	Inactive-mode enable.
PINAEX-	Results from the data of instruction EXC.
PINEXC-	Indicates the memory has entered the expansion mode by an interrupt.
PINRF +	Interrupt response indicating the load read-back operation complete.
PINRXE +	Interrupt enabler.
PIOEKn + (0-3)	I/O-error key bits.
PIOER +	I/O data-transfer error.
PIOERE +	I/O-instruction error enabler.
PIOUER +	I/O unassigned error.



Mnemonic	Description	Mnemonic	Description
PMAR C_n - (1,2)	Internal clock for map-address register.	PPMAWR-	PMA writing request.
PMD X_n - (0-17)	Expansion-memory data.	PPR n F + (1-4)	High-speed DMA priority lines.
PMDXEN-	Enabling signal for PMD X_n - (0-17).	PPRK Y_n + (0-3)	Output bits from the memory map's key register.
PMFCMP +	Clock for error testing of halt, jump, and I/O instructions.	PPRMF +	Priority input signal for high-speed DMA.
PMHG n E + (1-3)	Enabling signals for MHG Y_n - (1-3).	PPRMR +	Indicates processor is requesting memory.
PMHM n E +	Enabling signals for MHM Y_n - (1-3).	PPRMX +	System interrupt priority input.
PMIRBF +	Buffered MIRAB +.	PPRMXB +	Buffered PPRMX +.
PMODSW +	Memory-map mode switching.	PPRN X -	System interrupt priority output.
PMPEN +	Enabling signal for the memory protection function.	PPROWR-	Processor writing request.
PMRBPE +	Enables the memory request to by-pass the inactive memory map.	PR n X- I (1-9)	I/O-bus priority lines. PR1 X - I is the highest priority and PR9 X - I is the lowest.
PMRQ n E + (0-3)	Enabling signals for MRQ Y -, MRQ Y 1 +, MRQ Y 2 +, and MRQ Y 3 +.	PRAM n - (0-7)	Output bits from the RAM address multiplexor.
PMRQD1 +	Delayed memory request output 2.	PRAM S_n - (0-3)	Row selectors for the RAM array.
PMRQOI-	Memory request output inhibitor.	PRAMW +	When high, loads data into the RAM array. When low, enables addressed data to be read out of the RAM array.
PMRQOT +	Memory request output.	PRDI n - (11-12)	Swapping control bits.
PMRQPA +	Clock for testing unassigned address, instruction fetch, and writing and I/O data-transfer errors.	PRDMEX-	Resets the memory map's DMA-transfer logic.
PMRQYD +	Delayed memory request output 1.	PRDO n (0-12)	Output data from RAM array.
PMRYDN +	Goes true when memory acknowledgment and memory request are true.	PRIVLG +	Enabling signal for privileged instructions.
PMSWIP +	Indicates the memory map is in the process of switching from one operating mode to another.	PRMA K_n + (0-3)	Input for key multiplexor from either the PMA option or the memory map's key register.
POINTE +	Enabling signal for memory-protection internal interrupt.	PRMWEN +	Enabling signal for PRAMW +.



MNEMONICS

Mnemonic	Description	Mnemonic	Description
PROPRD +	Indicates that only operand fetches are permitted.	PWIOC-	Loads the initial map address into the map-address counter.
PRWRn- (11,12)	Read/write control signals for bits 11 and 12 of the RAM array.	PWRER-	Indicates a writing error.
PSDMEX-	Starts a memory-map DMA transfer.	PWRFUA-	Writing into a full-access page.
PSEDMA +	Senses if the memory map is performing a DMA operation.	PWRKC-	Write-key control signal. Loads data into key register, 64K-memory register, and executive-state register.
PSEn + (0,1)	Decoding bits for a SEN instruction.	PWRT-	Writing request.
PSEnEN +	Enabling signal for SEN instruction.	PXFRW-	Loads data into the word-transfer counter.
PSIn + (0,1)	Selector signals for the I/O-bus data multiplexor.	PYD123 +	Memory acknowledgment from expansion memory 1, 2, or 3.
PUADn + (0-15)	Output bits from unassigned address register.	PYDNMD +	Delayed memory acknowledgment from expansion memory 1, 2, or 3.
PUASER +	Indicates an unassigned address error.	SERX- I	I/O-bus sense response.
PUNASG +	Indicates an unassigned page.	SPFA-	System power failure alarm.
PUSAGW +	Writing strobe for usage bit.	SRST-	System reset.
PUSERM +	User mode.	SYRT- I	I/O bus system reset.
PUSRE +	Enabling signal for user mode.	TPIF- I	High-speed trap-in request of I/O bus.
PUSREX-	Switches memory map from executive mode to user mode.	TPOF- I	High-speed trap-out request of I/O bus.
PWCRCn- (1,2)	Internal (ripple) clock for word-transfer counter.	YDNMA +	Memory acknowledgment from port A of mainframe memory.
PWCZ-	Output of word-transfer counter.	YDNMB +	Memory acknowledgment from port B of mainframe memory.
PWIMA-	Loads the initial memory address into the DMA memory-address counter.	YDNMn + (1-3)	Memory acknowledgments from expansion memories.



SECTION 7

TEST PROGRAMS

7.1 GENERAL

The memory map test program, which is controlled by the MAINTAIN II test executive program, is used to verify correct operation and to isolate malfunctions. Minimum hardware requirements for using the test program consists of a Varian 70 series processor with 8K of mainframe memory (core or semiconductor), a memory map, and an ASR-33 Teletype (TTY). For a more efficient operation, a high-speed paper tape reader/punch is recommended.

The memory map test program resides primarily in the second 4K increment of physical memory. However, part of the program, along with the test executive, resides in the first 4K increment.

In this section, a number with a leading zero is octal and one without a leading zero is decimal.

7.2 TEST PROGRAM ORGANIZATION

The test program consists of the following six components:

- Miscellaneous Register Test
- Map-Register Test
- Executive-State Access and Map-Selection Test
- Memory Test
- Memory Protection Test
- Utility Routines

The first two tests are executed with the memory map in the inactive mode.

7.2.1 Miscellaneous Register Test

This test sets and verifies all states of the key, 64K-memory, and executive-state registers. Each state of the registers are established via a write-key control function of an output-data transfer. Verification of the register contents is made by reading the status word of an input-data transfer (figure 3-2).

7.2.2 Map Register Test

In this test, the RAM array and DMA control circuits of the memory map are tested. There are five subtests contained in this test:

- a. Unique address test: This test verifies the existence of all 1024 locations of the RAM array. Using a single DMA block transfer, all of the 13-bit locations are loaded and read back. Each location is loaded with its own address. For example, location 0 should contain all zeros and location 1023 should contain an octal 01777 (1023 in decimal).

- b. Alternate ones (012525) test: This test stores alternate ones (octal 012525) in each of the 1024 locations of the RAM array. It verifies that both the zero and one states exist for each bit of each location.
- c. Alternate ones (005252) test: This test uses the complement pattern (005252) of the second test for testing each location of the RAM array.
- d. Worst-case chip test: This test "walks" a one-bit through zero bits and then a zero bit through one bits for each of the four rows in the RAM array. These tests are performed on all 64 locations in maps 0, 5, 10, and 15 of the RAM array.
- e. Worst-case RAM test: This test "walks" a word of ones through all-zero locations of the RAM array, and similarly "walks" a word of zeros through all-one locations. The sequence for walking a word of zeros through the RAM array is as follows:

1. Load all ones into all locations of the RAM array.
2. Load all zeros into the first location of the RAM array.
3. Verify the contents of the RAM array.
4. Repeat the above sequence incrementing the address of the all-zero word until all 1024 locations of the RAM array are tested.
- f. Selective map addressing test: This test verifies that various block lengths and starting addresses can be properly addressed. The RAM array is loaded with a unique address pattern. Each location is read and compared with its expected unique address. Each block length is different and corresponds to the starting address of the respective block, for example: 1, 2, 4, 8, 16, 32, . . . 512.

7.2.3 Executive-State Access and Map-Selection Test

Fetch and store operations for the four states of the executive mode are tested. In addition, the test verifies that correct reading and writing operations can occur with the user maps.

7.2.4 Memory Test

Before the memory test is run, a unique-bit test is performed on the 16 data lines of the mainframe and expansion memory buses. The memory test runs a unique-address test to verify that a unique address exists for each physical memory location above the first 8K of mainframe memory (bus 0). If only 8K of memory is used, this test is bypassed. Each memory location is loaded with its own address. A reading and comparing process is then performed on each address. In addition, all-ones and all-zeros tests are performed.



TEST PROGRAMS

A special subtest tests the swapping control bits (change and usage bits).

By using 64K of memory (in addition to the 8K used by the test programs) all memory map hardware can be tested by moving the memory to the various memory buses (0 through 3).

7.2.5 Memory Protection Test

The memory protection test, which consists of 25 subtests, verifies that the memory map can prevent specified operations from occurring. Refer to section 3.6 for a description of the memory protection function.

7.2.6 Utility Routines

The utility routines are aids in troubleshooting and in establishing memory-map conditions for the running of other test programs. The following utility routines are provided:

- Dump the contents of a map. The user specifies one of 16 maps for a TTY print-out. The starting map address and number of words (1 to 64) are also specified.
- Change a data word of a map. The user specifies the map, the map address, and the new data.
- Load map. This routine reads a paper tape (BLD format) and transfers the data to a user specified map.
- Set BIC key register. The user specifies the BIC device address and key bits.
- Set BTC key register. The user specifies the BTC device address and key bits.
- Set memory-map key register. The user specifies the key bits.
- Switch to user mode. This routine switches the memory map from the executive mode to the user mode.

7.3 PROGRAM-TAPE LOADING

Before loading the memory-map test program tape, the MAINTAIN II test executive tape must be loaded into the reader (refer to Test Programs Manual, 98 A 9952 06x). Included in the test executive is the object tape loader for the memory map test program. After the test executive is loaded, mount the object tape of the memory map test program in the reader. The memory map test program tape is identified by the Varian part number 92U0109-002x punched in the leader portion of the tape. Position the tape in the reader past this area. To load and start the execution of the test program, use the TTY to type an L followed by a period.

7.4 SENSE SWITCHES

Operation of the memory map test program can be modified by setting and resetting SENSE switches on the

computer control panel. The switch functions are listed in table 7-1.

The test executive contains power failure/restart routines that save and restore the A, B, and X registers. However, the test program does not restore the volatile contents of the memory map RAM array. Consequently, if a power failure occurs when the memory map is active the test program will fail.

Table 7-1. SENSE Switch Settings

SENSE Switch	Set	Reset
1	Suppresses error messages	Prints error messages
2	Halts program on error (see note 1)	No halts on error
3	Terminates the test and returns program to start of test (see note 2)	Continues test

Note 1: After the error halt, one of the following operations can be performed:

- a. To allow the program to continue to the next error halt, keep SENSE switch 2 set and press START.
- b. The program can be made to loop on the error condition, by resetting SENSE switch 2 and pressing START. The program continues to loop until SENSE switch 2 is set; it then continues to run in the halt-on-error mode.

Note 2: In the memory test, the setting of SENSE switch 3 terminates extended error messages and causes the program to loop within the memory test. The memory test continues to run when SENSE switch 3 is reset.

7.5 OPERATING PROCEDURES

When loading is completed, the test program is automatically directed to the starting memory address 0500, and the TTY prints the message:

```
THIS IS THE MEMORY MAP TEST
**
```

The double asterisk indicates the program is waiting for an input directive. To initiate the five tests of the test program, the operator must type a period; or, to initiate a utility routine, the operator must type the appropriate utility directive (section 7.6).

When tests are initiated, the TTY prints a request for input parameters. The following is a TTY printout requesting the parameters (the underlined characters are provided by the



operator, and are terminated with a period unless otherwise specified):

```
MEM CONFIG = a,b,c,d.
TEST SEQ = i,i,--i.
CYCLES = n.
```

where :

a, b, c, and d are octal numbers specifying the amount of physical memory (0 to 64K) on buses 0, 1, 2, and 3, respectively. All four parameters must be specified initially, and each can have a value in the octal range 0 through 100. After the parameters have been initially specified and the program has returned to this point, a period typed by the operator causes the program to use the previously designated memory values.

i,i-i can either be a period specifying that all tests are to be executed in the normal sequence, or up to five numbers (1 through 5) specifying the desired sequence of executing the tests. Test number assignments are listed in table 7-2.

n is an octal number specifying the number of cycles the test is to run. The terminator can be either a comma or a period. If a comma is used, a message is printed indicating a cycle of the test is completed. If a period is used, the message is not printed. A terminator with no number specifies continuous execution of the program. If a comma is used, the TTY prints the following message upon completion of a test cycle:

```
END CYCLE n
```

where **n** is an octal number indicating the cycle just completed.

Upon completion of the specified number of cycles or upon termination via SENSE switch 3, the following end-of-test message is always printed:

```
TEST COMPLETE
TOTAL CYCLES = n
```

where **n** is an octal number indicating the total number of cycles executed.

Following the above message, the TTY prints the double asterisk indicating the program is waiting for a new input directive.

Table 7-2. Test Number Assignments

Number	Test
1	Miscellaneous register test
2	Map-register test
3	Executive-states access and map-selection test
4	Memory test
5	Memory protection test

7.6 INPUT DIRECTIVES FOR UTILITY ROUTINES

The following input directives are required to initiate the utility routines:

Input Directive	Description
DM m, r, n.	Dump the contents of a map. m specifies the map and can be any number 0 through 017. r specifies the starting address and can be any number 0 through 077. n specifies the number of words and can be any number 0 through 077. If r+n is greater than 077, the dump (TTY print-out) is terminated after the last word of the specified map is read. An entire map can be dumped by entering only the m parameter.
CM m, n, x.	Change a data word of a map. m specifies the map, n specifies the word, and x specifies the new value. If a comma is entered as the terminator instead of a period, successive words of the map can be changed. Incrementing past word 077 is not allowed and causes the routine to terminate.
LM m.	Load a map. m specifies the map. The data are read from a paper tape (BLD format) and transferred to locations 0 through 077 of the specified map. Physical memory locations 01000 through 01077 must be used for generating such object tapes. The test executive control statements C and P can be used to establish the data in those locations and then punch these data onto paper tape. The execution address for punching the tape is 044.
BIC a, x.	Set BIC key register. a specifies the device address and x specifies the key-register value.
BTC a, x.	Set BTC key register. a specifies the device address and x specifies the key-register value.
MK x.	Set memory-map key register. x specifies the key-register value.
UMI.	Switch to user mode. I specifies a location in the user map. The contents of the memory-map key register specifies the user map.

7.7 ERROR MESSAGES

An illegal operator input causes the TTY to print the double asterisk indicating the program is waiting for a new input.



TEST PROGRAMS

The following error message, which is associated with the load-map utility routine, occurs if a check-sum error is detected when reading the paper tape:

CHECKSUM ERROR = aaaaaa

where **aaaaaa** is the address of the record containing the check-sum error.

Because DMA transfers occur throughout the program, the following error messages may be associated with any of the five test components:

SENSE DMA BUSY TIME OUT

Each of the five test components has a basic error message format. The first field of each error message is a test identifier T1---T5. For test components 1, 2, and 4, each remaining field is prefaced with an identifying letter as follows: R = location in the specified map, M = map, E = expected test data, A = actual test data, P = physical memory address, and V = logical (virtual) memory address. Error messages are summarized in section 7.8.

7.7.1 Miscellaneous Register Test

When an error is detected in this test, the TTY prints the following:

T1 R-r E-eeeeee A-aaaaaa

where:

r = K, VE, or ES identifying one of the following memory map registers: key, 64K-mode, or executive-state.

eeeeee is the expected register value.

aaaaaa is the actual register value.

The key and executive-state registers have four significant digits; the 64K-mode register has one.

7.7.2 Map Register Test

When an error is detected in this test, the TTY prints the following:

T2 M-n R-mm E-eeeeee A-aaaaaa

where:

n is the key number (0 through 017) of the map containing the error.

mm is the location (0 through 077) in map **n** containing an error.

eeeeee is the expected value contained in the map location.

aaaaaa is the actual value contained in the map location.

The map-location values are in the range 0 through 017777 to cover the 13 bits of a RAM-array location.

7.7.3 Executive-State Access and Map-Selection Test

When an error is detected in this test, the TTY prints the following:

T3 ERROR = t, n

where:

t = 0 through 3 identifies the executive mode state under test.

t = 4 indicates map selection testing.

n is the type of error.

The following is a list containing the types of errors for the executive mode testing:

n	Type of Error
1	Operand fetch was not from map 0
2	Operand fetch was not from map 1
3	Operand store was not from map 0
4	Operand store was not from map 1

For the map-selection testing, **n** can be either 01, 02, 04, or 010 indicating that one of these four maps are being tested.

If the instructions comprising this test are not fetched from map 0, the results are unpredictable. The operand store testing assumes that the operand fetches are operating correctly.

7.7.4 Memory Test

When an error is detected in this test, the TTY prints the following:

T4 M-n P-p,ppppp V-vvvvvv E-eeeeee A-aaaaaa

where:

n is always 0 because the memory test must be executed only in map 0 to facilitate error reporting.

p,ppppp is the 18-bit physical address where the error occurred. The digit before the comma contains the three high-order bits, and the digits to the right contain the remaining 15 bits.

vvvvv is the 15-bit logical (virtual) address where the error occurred.

eeeeee is the expected 16-bit value contained at the memory address.

aaaaaa is the actual 16-bit value contained at the memory address.



If a DMA busy time-out occurs when reading the map, Xs are printed in place of the physical-address digits.

For the swapping control subtest, the error message is modified as follows:

- a. The physical address contains Xs for the low order positions (to the right of the comma). These Xs have no meaning.
- b. The logical-address print-out consists of either the characters C-BIT or U-BIT. The characters C-BIT and U-BIT indicate an error in the change bit and usage bit, respectively.
- c. The expected and actual address contents are either 000000 or 000001.

7.7.5 Memory Protection Test

When an error is detected in this test, the TTY prints the following:

T5 MP ERROR = m, n

where

m is a key number (0 or 05) of the map containing the error.

n is a memory protection error code as described in table 7-3.

An interrupt to an incorrect address halts the program if the incorrect address is one of the memory-map interrupt addresses 020 through 035 (table 3-4). For each memory protection subtest, the valid (or expected) interrupt location is loaded with a jump instruction for normal processing. The invalid locations 020 through 035 contains a halt code.

The memory protection test also verifies that an interrupt occurs when all DMA transfers are completed in a memory map loading or read-back operation. If the correct interrupt to address location 016 is not received, the TTY prints the following message:

T5 NO DMA TRANSFER COMPLETE INTERRUPT

Note: When used with VORTEX, the memory map is wired to inhibit the DMA transfer-completion interrupt. Consequently, this error message is normal (or expected) for VORTEX systems.

Table 7-3. Memory Protection Error Descriptions

Error Code	Test Description	Expected Result
1	Execute a halt instruction.	Interrupt to location 020.
2	Execute an I/O instruction (OAR).	Map 0: No interrupt Map 5: Interrupt to location 022.
3	Write into a read-operand only location.	Interrupt to location 024.
4	Determine if writing cycle of subtest 3 was changed to a reading cycle to prevent memory alteration.	No change in contents of the location where the writing cycle was directed.
5	Determine if the instruction address register is tracking the program counter.	Instruction address register contains the location where the error occurred plus one (contents of program counter plus one).
6	Disabled memory protection and perform a writing violation.	No interrupt.
7	Determine if writing cycle of subtest 6 was inhibited.	No change in contents of the location where the writing cycle was directed.
10	Re-enable memory protection and write into a read only location.	Interrupt to location 024.

(continued)



Table 7-3. Memory Protection Error Descriptions
(continued)

Error Code	Test Description	Expected Result
11	Determine if writing cycle of subtest 10 was changed to a reading cycle to prevent memory alteration.	No change in contents of location where the writing cycle was directed.
12	Jump to a read-operand only location.	Interrupt to location 026.
13	Read an unassigned location.	Interrupt to location 030.
14	Determine if the unassigned address register contains the unassigned address for a reading error.	Unassigned address register contains the logical address where the error occurred.
15	Write into an unassigned location.	Interrupt to location 030.
16	Determine if the unassigned address register contains the unassigned address for a writing error.	Unassigned address register contains the logical address where the error occurred.
17	Determine if the writing cycle of subtest 15 was changed to a reading cycle to prevent memory alteration.	No change in contents of location where the writing cycle was directed.
20	Fetch instruction from read-operand only location.	Interrupt to location 032.
21	DMA output data is transferred to an unused map. Transfer to be from locations in an unassigned page.	Interrupt to location 034.
22	Tests the control logic for the abnormal DMA termination (SEN 0146) and the DMA transfer reset (EXC2 0446).	Upon entry from test 21, sense DMA activity = no and sense abnormal DMA termination = yes, and instruction EXC2 0446 resets the sense status of the abnormal DMA termination.
23	Read the status word and check the I/O error key bits, unassigned/write error bit, and PMA/DMA error bit.	I/O error key bits = 0. Unassigned/write error bit = 1. PMA/DMA error bit = 1. See note.
24	DMA input-data transfer. Read part of a map and store into a read only page.	Interrupt to location 034.
25	Read the status word and check the I/O error key bits, unassigned/write error bit, and PMA/DMA error bit.	I/O error key bits = 0. Unassigned/write error bit = 0. PMA/DMA error bit = 1. See note.

(continued)



Table 7-3. Memory Protection Error Descriptions
(continued)

Error Code	Test Description	Expected Result
26	Set executive mode to state 2. Switch to map 5 and execute a halt instruction. Determine executive state when interrupt occurs by doing an executive-state operand fetch.	Interrupt to location 020. Executive-mode state = 0.
27	Execute EXC2 546 to remove the executive mode mask. Determine the executive-mode state.	Executive mode state = 2.
30	Read (LDAE) locations 050000 and 077777 which are in unassigned pages.	Interrupt to location 030 on both reading cycles. Unassigned address register contains 050000 and 077777, respectively, upon detection of the error interrupts.
31	Execute halts from logical memory locations 077776 and 077777.	Interrupt to location 020 for each halt. The instruction address register contains 077777 following the halt at 077776 and contains 000000 following the halt at 077777.

Note:

In normal system operation, the I/O error key bits would contain the user key number that caused the error. This test uses the executive mode's DMA facility for I/O data transfers.

7.8 SUMMARY OF I/O MESSAGES

The following is a list of general operating messages for the memory map test program:

- THIS IS THE MEMORY MAP TEST
- **
- MEM CONFIG =
- TEST SEQ =
- CYCLES =
- END CYCLE n
- TEST COMPLETE
- TOTAL CYCLES = n

The following is a list of input directives for the utility routines:

- DM m, r, n. Dump the contents of a map.
- CM m, n, x. Change a data word of a map.
- LM m. Load a map.
- BIC a, x. Set BIC key register.
- BTC a, x. Set BTC key register.
- MK x. Set memory-map key register.
- UMI. Switch to user mode.

The following is a list of error messages:

- CHECKSUM ERROR X = aaaaaa
- SENSE DMA BUSY TIME OUT
- T1 R-r E-eeeeee A-aaaaaa
- T2 M-n R-mm E-eeeeee A-aaaaaa
- T3 ERROR = t, n
- T4 M-n P-p,pppppp V-vvvvvv E-eeeeee A-aaaaaa
- T5 MP ERROR = m, n
- T5 NO DMA TRANSFER COMPLETE INTERRUPT

7.9 MAINTAIN II OPERATION WITH MEMORY MAP

The standard MAINTAIN II test programs can be run in a system with the memory map either inactive (unmapped environment) or active (mapped environment).



TEST PROGRAMS

7.9.1 Unmapped Environment

The memory map can be placed in the inactive mode by pressing the RESET switch on the computer control panel. In this mode, the first 32K of mainframe memory (64K if the writable control store is used) is available to the processor unmapped and all instructions are permitted. By following the procedures in the Test Programs Manual, 98 A 9952 06x, the tests programs can be loaded and executed.

7.9.2 Mapped Environment

The following procedures are to be used when loading and executing the test programs in a mapped environment:

- a. Load the test executive program (refer to Test Programs Manual).
- b. Load the memory map test program (section 7.3).
- c. Execute the memory map test program to verify the memory map operation (section 7.5).
- d. Use either the change map data or load map utility routine to establish the contents of a map and equate the desired physical memory to logical memory. Any user map number 0 through 017 can be used. The memory-map's memory protection function is disabled so that I/O instructions operate normally. **Caution:** All maps must include the first 4K of physical memory in which the test executive and MAINTAIN II test programs reside. Section 7.9.3 contains an example in which a paper tape of a map is used to establish a map.
- e. Use the set memory-map key utility routine to select a map.
- f. Use either the set BIC key or set BTC key utility routine to select a map.
- g. Use the switch to user mode utility routine to transfer to location 07000 in the selected map. Location 07000 is the normal starting address for the test executive.
- h. Initiate execution of the test executive and load in the MAINTAIN II test program using procedures in the Test Programs Manual.

The above steps complete the procedures for establishing a mapped memory and executing a test program in the mapped system. Steps b through h must be repeated each time a test program is run in a different mapped configuration.

7.9.3 Example

In the following example, the MAINTAIN II memory test (part 2) is executed in a map containing the first 8K of memory on bus 0 plus the first 24K of memory on bus 1.

- a. Load the test executive program.
- b. Using the change (C) control statement of the test executive, store the following data in memory locations 01000 through 01077: 001000 through 001017 and 001200 through 001257. Setting bit 9 of each data word equal to one provides the full access mode of operation (section 4.3.18). Setting bit 7 of data words 001200 through 001257 to one selects the first 24K of memory on bus 1.
- c. Using the punch (P) control statement of the test executive, punch the map contents onto paper tape via the TTY. The control statement must be: P1000, 1077, 44.
- d. Load the memory map test program and let it execute for one cycle. The following is an example of the TTY message for this example (the print-out for the memory configuration assumes 32K on buses 0 and 1, and no memory on buses 2 and 3):

```
THIS IS THE MEMORY MAP TEST
**
```

```

•
MEM CONFIG = 040,040,0,0.
TEST SEQ = .
CYCLES = 1.
```

```
TEST COMPLETE
TOTAL CYCLES = 000001
**
```

- e. Using the load map utility routine, load data from the paper tape into map 2. The input directive is:

```
LM2
**
```

- f. Using the utility routine for setting the memory-map key register, set the key register for map 2. The input directive is:

```
MK2
**
```

- g. Using the utility routine for switching to the user mode, switch to map 2 which will contain the test executive in the first 4K of memory on bus 0. The input directive is:

```
UM7000
```

- h. By pressing START on the computer control panel, start the execution of the test executive. The TTY then prints:

```
THIS IS THE 620 TEST EXECUTIVE
MEMORY SIZE IS 32K
```

- i. Load and execute the MAINTAIN II memory test using the procedures in the Test Programs Manual.



7.10 TTY MESSAGE EXAMPLES

The following is a TTY print-out from the execution of a memory map test program with 32K of memory on bus 0 and no memory on buses 1, 2, and 3:

THIS IS THE MEMORY MAP TEST

```

**
.
MEM CONFIG = 40,0,0,0.
TEST SEQ = .
CYCLES = 2.
END CYCLE 000001

```

END CYCLE 000002

```

TEST COMPLETE
TOTAL CYCLES = 000002

```

**

The following is a TTY print-out with the same memory configuration as in the previous example, but the user has specified an irregular sequence for executing the test components:

THIS IS THE MEMORY MAP TEST

```

**
.
MEM CONFIG = 40,0,0,0.
TEST SEQ = 5,4,3,2,1.
CYCLES = 1.
TEST COMPLETE
TOTAL CYCLES = 000001

```

```

**
.
MEM CONFIG = .
TEST SEQ = 5,4,3,2,1.
CYCLES = 1,
END CYCLE 000001

```

```

TEST COMPLETE
TOTAL CYCLES = 000001

```

**

The next TTY print-out shown is a dump of map 11 using the dump utility routine. The map is set up for the first 8K of memory on bus 0 plus the first 24K of memory on bus 3. All pages are set for the full-access mode of operation.

**

DM11.

```

( 000000 ) 005000 005001 001002 005003 005004 001005 001006 005007
( 000010 ) 001010 001011 001012 001013 001014 001015 001016 001017
( 000020 ) 005600 005601 005602 005603 005604 005605 005606 005607
( 000030 ) 005610 005611 005612 005613 005614 005615 005616 005617
( 000040 ) 005620 005621 005622 005623 005624 005625 005626 005627

```

(continued)



TEST PROGRAMS

```
( 000050 ) 005630 005631 005632 005633 005634 005635 005636 005637
( 000060 ) 005640 005641 005642 005643 005644 005645 005646 005647
( 000070 ) 005650 005651 005652 005653 005654 005655 005656 005657
**
```

Using the utility routine for changing a map data word, map 11 in the previous example can be modified. The following TTY print-out shows input directives and the dump of the modified map 11:

```
**
CM11,10,5657.
**
CM11,77,15010.
**
CM11,37,5650.
**
CM11,70,5617.
**
DM11.
( 000000 ) 005000 005001 001002 005003 005004 001005 005006 005007
( 000010 ) 005657 001011 001012 001013 001014 001015 001016 001017
( 000020 ) 005600 005601 005602 005603 005604 005605 005606 005607
( 000030 ) 005610 005611 005612 005613 005614 005615 005616 005650
( 000040 ) 005620 005621 005622 005623 005624 005625 005626 005627
( 000050 ) 005630 005631 005632 005633 005634 005635 005636 005637
( 000060 ) 005640 005641 005642 005643 005644 005645 005646 005647
( 000070 ) 005617 005651 005652 005653 005654 005655 005656 015010
```

The following TTY print-out shows the execution of the MAINTAIN II memory test (part 2) from map 11 of previous example (note the memory errors detected):

```
**
MK11.
**
UM7000.
THIS IS THE 620 TEST EXECUTIVE
MEMORY SIZE IS 32K

L.
MEMORY TEST
MEMORY SIZE IS 32K

4K MODULE(S) TO BE TESTED =.
CYCLES = 1.
TEST    ADDRESS  EXPECTED  ACTUAL
000001  024022    024022    026022
000005  022626    177777    177377
000005  022626    177777    177377
000005  062251     000000    010000
000005  062251     000000    010000
000005  062251     000000    010000
ERROR TOTAL = 000006
NUMBER OF CYCLES RUN = 000001
MEMORY SIZE IS 32K
```

The next TTY print-out shown is from the paper tape and BIC test program executed from map 2. With the load map utility routine, map 2 is loaded from a paper tape containing data for mapping 8K of memory on bus 0 and 24K of memory on bus 3.



THIS IS THE MEMORY MAP TEST

**

.

MEM CONFIG = 30,0,0,30.

TEST SEQ =.

CYCLES = 1.

TEST COMPLETE

TOTAL CYCLES = 000001

**

LM2.

**

MK2.

**

BIC20,2.

**

UM7000.

THIS IS THE 620 TEST EXECUTIVE

MEMORY SIZE IS 32K

L.

620 PAPER TAPE AND BIC TEST

PT PUNCH DA = 37.

PT READER DA = 37.

BIC TEST REQUESTED?

Y

BIC DA = 20.

PIM USED?

N

CYCLES = 2.

000000 ERROR(S)

BIC TEST REQUESTED?



varian data machines

TITLE _____
MANUAL NUMBER _____

1. Please complete the following chart.

[illegible]

☐ Below Average

d. NAME _____

Fold

FIRST CLASS
PERMIT NO. 323
NEWPORT BEACH,
CALIFORNIA

BUSINESS REPLY MAIL

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES



varian data machines / a varian subsidiary
2722 michelson drive / irvine / california / 92664

ATTN: TECHNICAL PUBLICATIONS

Fold

Staple



varian data machines / a varian subsidiary